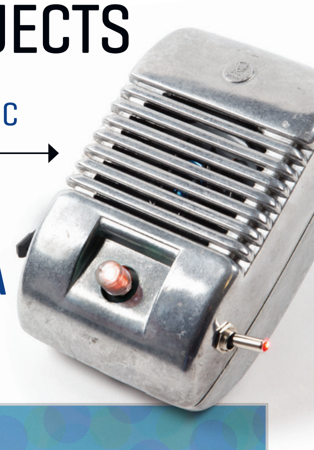
HARRY POTTER ELECTRONICS WAND // 25+ PROJECTS

Make:

- Supercharged Classic DRIVE-IN SPEAKERS
- LED Matrix Handbag
- GIF-MAKING CAMERA
- Tie-Dye Flowers



DIY ROBOTS!

4 FUN BOTS TO BUILD:

CHIP-E
THIS BIPEDAL ROBOT HAS MOVES!

+

FPV BALANCE BOT

8-LEGGED LEGO WALKER

MINDSTORMS SOUND SEQUENCER

+OUR GUIDE TO **25 ROBOT KITS**

LEARN SKILLS!

USING MULTIMETERS

PROGRAMMING ROBOTS

HAND STITCHING LEATHER

CONTENTS

Make: Volume 55 February / March 2017

Bot Factory
SPECIAL SECTION

40

ON THE COVER: Two dancing Chip-E bots stretch before their big show. Photo: Hep Svadja.

Rosa de Jong, Hep Svadja

COLUMNS

FEATURES

SKILL BUILDER

PROJECTS

FUN ZONE

TOOLBOX

SHOW & TELL

Make:

> "... you just can't differentiate between a robot and the very best of humans." *— Isaac Asimov, I, Robot*

PUBLISHED BY

MAKER MEDIA, INC.
Dale Dougherty

Comments may be sent to:
editor@makezine.com

Visit us online:
makezine.com

Follow us:
@make @makerfaire @makershed
google.com/+make
makemagazine
makemagazine
makemagazine
twitch.tv/make
makemagazine

Manage your account online, including change of address: makezine.com/account
866-289-8847 toll-free in U.S. and Canada
818-487-2037, 5 a.m.–5 p.m., PST
cs@readerservices.makezine.com

CONTRIBUTORS

For many, automated machines are part of daily life, from thermostats to self-parking cars. So what then defines a robot versus not-a-robot?

Renee L. Glinski
Houston, TX (EddiePlus Self-Balancing Robot)
To me a robot is a collection of components programmed to work together to perform a function and re-programmable (self or other-wise) to perform another.

Ben Vagle
Denver, CO (TrotBot)
A robot is a machine that is capable of performing complex tasks autonomously, while a non-robot requires a much higher degree of human intervention to perform such tasks.

Erica Tiberia
Toronto, Ontario, Canada (My Mini Mars Rover)
A key part is that a robot usually interprets some kind of input from its environment using a program to control an output, next step, or action.

Debra Ansell
Los Angeles, CA (LED Matrix Handbag 2.0)
The difference between a robot and not-a-robot is that you don't try to fix a robot with percussive maintenance.

Damon McMillan
Sunnyvale, CA (The Little Boat That Could)
A robot interacts with its environment in a physical way. It must move, manipulate, or make.

Issue No. 55, February /March 2017. *Make:* (ISSN 1556-2336) is published bimonthly by Maker Media, Inc. in the months of January, March, May, July, September, and November. Maker Media is located at 1160 Battery Street, Suite 125, San Francisco, CA 94111, 877-306-6253. SUBSCRIPTIONS: Send all subscription requests to *Make:*, P.O. Box 17046, North Hollywood, CA 91615-9588 or subscribe online at makezine.com/offer or via phone at (866) 289-8847 (U.S. and Canada); all other countries call (818) 487-2037. Subscriptions are available for $34.99 for 1 year (6 issues) in the United States; in Canada: $39.99 USD; all other countries: $50.09 USD. Periodicals Postage Paid at San Francisco, CA, and at additional mailing offices. POSTMASTER: Send address changes to *Make:*, P.O. Box 17046, North Hollywood, CA 91615-9588. Canada Post Publications Mail Agreement Number 41129568. CANADA POSTMASTER: Send address changes to: Maker Media, PO Box 456, Niagara Falls, ON L2E 6V2

Why wasn't MakerBot in *Make:*'s 3D printer shootout?

A CURIOUS OMISSION ...

How is it that year after year *Make:* manages to do a comprehensive comparison of 3D printers — some of which I am familiar with and many I am not — yet time after time, there is no mention of MakerBot printers?

I'm curious because it's a known printer — available at schools, makerspaces, etc. — and one of the few I've seen in the wild and would have available to me to try something out.

It is my de-facto reference point and I'm sure I'm not the only one. In thinking about buying my own printer, I'd love to know how all these options stack up against it.

—Derek Graham, via email

Digital Fabrication Editor Matt Stultz Responds

Hey Derek, thanks for getting in touch. I'm in charge of *Make:*'s digital fabrication shootout. When we started our testing for the 2016 guide, there were a ton of new 3D printers coming to market so we decided to focus on only new or majorly upgraded machines. This left out MakerBot (and a bunch of other great machines like the Type A or Tinkerine Ditto) because the models available at the time hadn't changed from the previous year's testing.

Our first three issues of this annual special indeed included MakerBot products, with the Gen 5 Replicator being in our 2015 guide. In fact, when MakerBot did not send us a review unit that year, we bought one for inclusion.

That machine did well, but the model had a major problem that was outside the scope of our 3-day testing (although generally well known in the printing community). Those Replicators performed well right out of the box, but quickly degraded in quality because of issues with the smart extruder. You can read more about it here: makezine.com/go/makerbot-extruder-lawsuit.

MakerBot has since sent us their latest Replicator for review. Unfortunately, they did so too late to make this year's issue but they will be included in the online guide once my full round of testing (including extended testing) is complete.

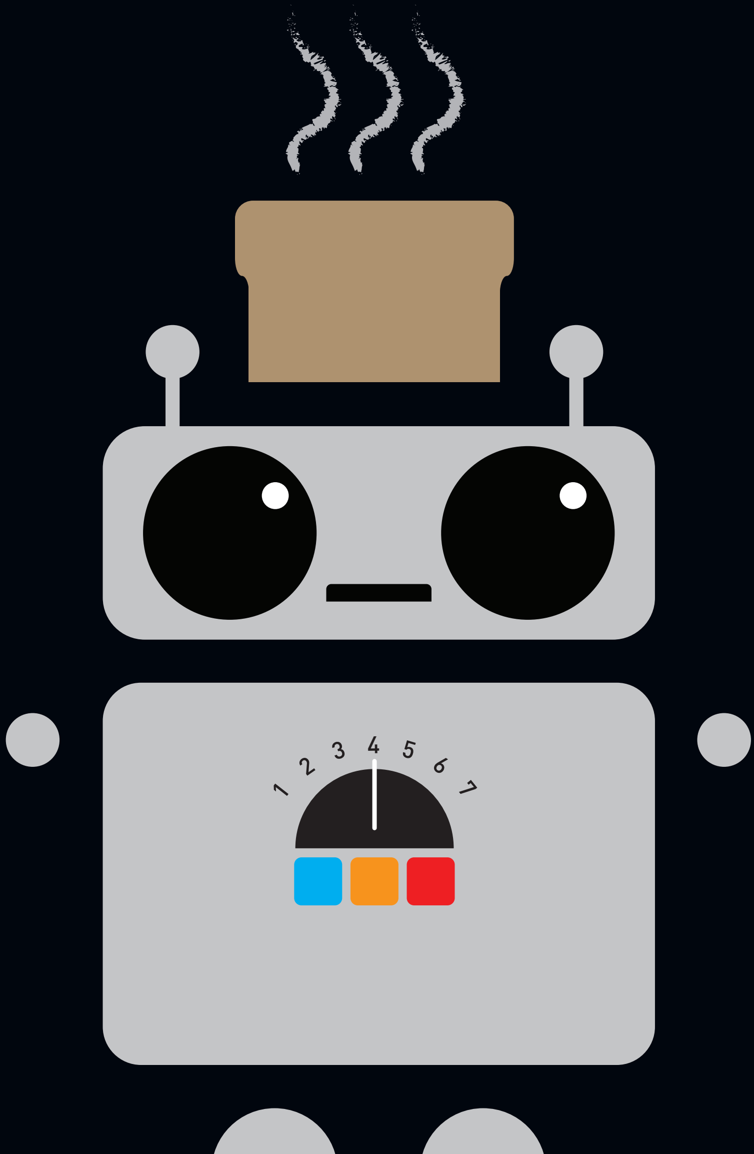

Christopher Garrison

MAKE AMENDS

In "Crystal Clear Ice Balls" (*Make*: Vol. 53), we omitted an author's name: Craig Belon, who helped pioneer the technique and coined the term "directional freezing." Learn more about the science behind this at makezine.com/go/directional-freezing.

Raising Robots

BY MIKE SENESE, executive editor of *Make:* magazine



Juliann Brown

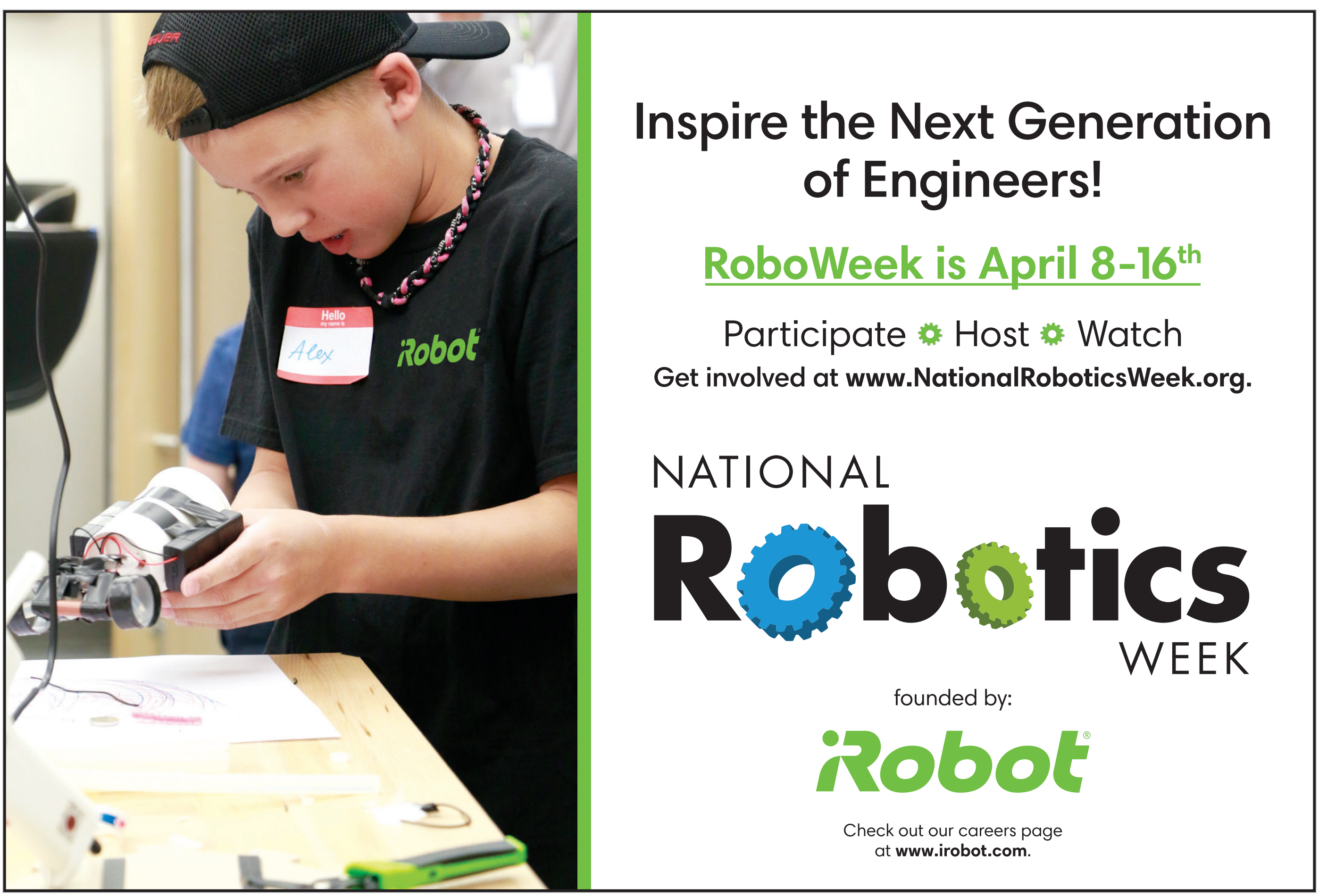
LAST YEAR, WHILE PREPARING FOR A PRESENTATION AT MAKER FAIRE KANSAS CITY, I dug through the *Make:* magazine back catalog to find our first mentions of key maker terms and concepts. With over 10 years of content chronicling the development of the maker world, this search turned into quite a surprising and enjoyable exercise. A few standouts I discovered: *Make:* first mentions Arduino, one of the most impactful developments in maker technology, in Vol. 5. 3D printing first appears in an article about the RepRap movement in Vol. 10 — a piece that actually doesn't include the words "3D printing." (That phrase first appears a year later, in *Make:* Vol. 14.) Drone shows up in Vol. 5, and Maker Faire (spelled "Fair") pops up early in Vol. 2.

"Robot," however, is a different case. As I searched through the archives, I realized that that word shows up in the first volume of *Make:* magazine — and then in every subsequent issue since. As intriguing as that is, it makes sense for us. Robot is a broad term that can be applied to almost any part of the technology world that we cover. It even extends beyond the typical uses that we associate the word with — as Zach Supalla puts it in *Make:* Vol. 52, "it's not a 'robotic bread warmer,' it's a 'toaster'."

We have an instinctual love for robots. It's something that's hard-coded into our DNA, a tricking of our reproduction instincts. Like the tigress in the zoo that raises piglets dressed in baby tiger outfits, our brains are stimulated by creating and nurturing elements that exhibit the ability to operate and think for themselves. From the simplest line-following bot, to the galloping creations from Boston Dynamics, to the Alexa-speaking Echo in my living room, to the neural network 1/10-scale car that navigated its way around Chris Anderson and Carl Bass' Self-Racing Car meetup track this past November, a tool that appears to know what it's doing is magical, even lovable, in an almost biological way.

Nearly all the terms I found in the *Make:* archives can be seen as parts of a robot ecosystem — the hardware brains, the fabricated body, the coding that processes sensor input and produces movement. We cover these elements in the robots section of this issue, and in some form, in every issue. As the tools have evolved, the capabilities of the projects our community and readers are making continue to blow our minds. We're happy to be a part of that path, and I can't wait to see what's next from here.

PCNC 440
PERSONAL CNC
smallsize
BIGRESULTS
FEATURES
Real Metal Cutting Power
Benchtop Size
42"x36" (WxD) Space Footprint
115V Power
10"x6.25"x10" (XxYxZ) Work Envelope
Includes CAD/CAM Package
Powered by PathPilot®
*Shown here with optional stands and accessories.
BASE PRICE
$4950
www.tormach.com
TORMACH.COM/440MAKERMILL
TORMACH®



Inspire the Next Generation of Engineers!
RoboWeek is April 8-16th
Participate Host Watch
Get involved at www.NationalRoboticsWeek.org.
NATIONAL Robotics WEEK
founded by:
iRobot®
Check out our careers page at www.irobot.com.

MADE ON EARTH

Backyard builds from around the globe

Know a project that would be perfect for Made on Earth?
Let us know: *makezine.com/contribute*

MAGICAL MICROCOSMS

BYROSA.NL

Minuscule houses appear to cling to tiny rock islands. Twigs transform into towering trees when juxtaposed with scrap-sized tents. Thread becomes the wire on petite power lines.

Each miniature building in **Rosa de Jong**'s *Micro Matter* collection is no bigger than the test tube they are suspended in — just a few centimeters wide at most. Yet they feel expansive, like you could get lost in every exquisite detail.

De Jong, a freelance designer, art director, and animator based out of Amsterdam, builds her creations from a combination of scavenged natural matter, such as rocks and plants, and modeling objects, like faux grass and moss. While she is careful not to reveal her process, her workspace is featured in her Instagram feed, offering glimpses of her carefully chosen items and partially finished projects.

Says de Jong on her website, "The most important thing for me is that by the end of a project, everyone is happy. I want to make things people want to look at, not something they're forced to look at." *—Lisa Martin*

See more of de Jong's work at makezine.com/go/test-tube-cities.

Rosa de Jong

CLEVER THINGS

FACEBOOK.COM/MAKEROLOGIST

The crew at Seattle-based **Makerologist** can now add interdimensional communication to their project portfolio. The group of six makers, lead by **Clarissa San Diego**, created this painstaking, 64-square-foot replica of Joyce Byers' living room wall from the TV series *Stranger Things*.

Faithfully recreating the show's iconic holiday light Ouija board was no simple task for the electronics team of **Micah Summers**, **Krunal Desai**, and **Katarina Wolcott**. Not content to take the easy route with addressable LED lights, the team individually wired period-accurate C9 incandescent bulbs to 26 DPDT relay switches passing live AC power. Two MCP23017 I2C expanders consolidated these connections down so that a single Arduino and a series of MOSFETS could manage them.

Light bulbs aren't the only detail they obsessed over. With the fabrication help of **Gabriel Bello-Diaz** and a steady soundtrack of '80s synth pop, the crew doctored up the perfect mix of worn wainscoting and dingy wallpaper to realize their vision for aesthetic accuracy.

To make the wall interactive, team member **Dan Halpern** leveraged the Bluetooth LE capability of the Arduino 101 board to allow for viewers to post messages to the wall using a mobile app. "We knew the installation was going to be in an open area where Wi-Fi would be unreliable," explains San Diego, "We created a mobile app with a simple interface of an input field and several buttons that would run natively from an iPad."

The project debuted at the 2016 Portland Mini Maker Faire, and was later exhibited at Seattle's EMP museum. So far, only a handful of visitors have been eviscerated by Demogorgons. *—Donald Bell*

Nat Seymour - Be Technical Graphics

Taekyeom Lee

ADOBE TYPE PORTFOLIO.TAEKYEOM.COM

A coil of perfectly smooth clay extrudes from **Taekyeom Lee**'s custom 3D printer, and a kiln-ready object begins to take shape on the printing bed. The striations of the layer lines add texture to the piece. Mistakes, sometimes encouraged by jostling the machine as it prints, give his work even more texture.

While others have built 3D printers capable of printing clay objects, Lee's machine and the artwork he creates with it are something special. The machine itself is based off of the delta RepRap printer. It is the result of months of obsessive iteration, which involved building the printer and fine-tuning the way the clay was extruded. Initially, Lee utilized a pneumatic pump to push out the clay, but this limited the size of the objects he could print. After researching methods for extrusion used in glue factories, Lee moved on to using a 3D printed auger valve paired with a stepper motor in order to extrude the clay.

The drive behind this whole project isn't just to explore 3D printing or ceramics, but to also explore typography. Lee is a graphic designer interested in taking letters out of the "glass box" of two-dimensional, high-contrast font to create 3D characters that have substance, texture, and even elements of interactivity. Letters transition from lowercase to uppercase, and short words are spelled, layer-by-layer, in clay.

Lee doesn't characterize himself as having any kind of special background in engineering or ceramics. As he puts it, "I begin with an ambitious idea and am willing to deal with endless troubleshooting." Much of his time spent working on this project was in testing different configurations for the printer and learning from failures and successes. His successes are stunning.

—Lisa Martin

See Lee's printer in action and learn more about his process at makezine.com/go/3d-printing-ceramics.

EXTRA FAST WHIMSY

INSTAGRAM.COM/SCOTT_BLAKE

After breaking his collarbone, artist and creative tinkerer **Scott Blake** was told not to ride a bike for a while. Rather than resting while he recovered, he set to work on an ambitious drill bike project.

Inspired by YouTuber Kim Henriksen, Blake spent about a month turning a children's Jamis bike and a couple of DeWalt cordless drills into a vehicle that can go over 20mph for about 5 miles on a single battery charge. "The first time I squeezed full throttle," says Blake, "it sent me straight on my back."

Because he used two drills in tandem, he had to install ratchet adaptors, "so that if one drill is spinning faster than the other, it'll just free-wheel. Sometimes one drill will just stop working, or get stuck, so it's nice that it can still keep going with just one." Blake also had to split the throttle cable and add a threaded rod to hold the drills in place.

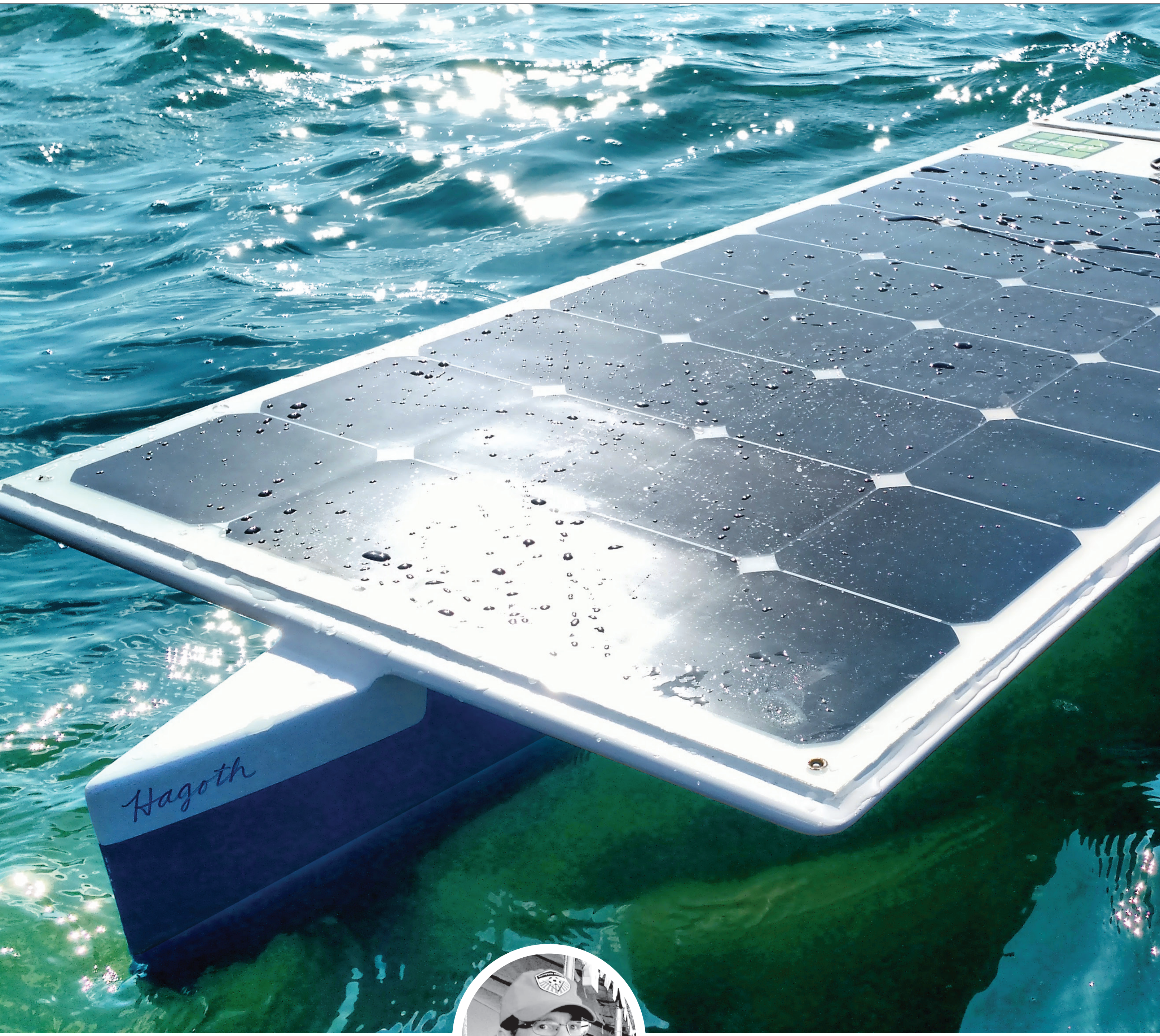
Blake kept customizing, using Collin Furze's "Motorhorse" and the miniature cars driven by the Shriners as inspiration. He hopes to ride the pony in parades.

"Adding the horse makes people smile," said Blake, "it makes me smile."

He installed LED lights inside the horse body, and is working on a costume with an oversized cowboy hat to cover his helmet. When asked what other customizations he had in mind for his motorized creation, Blake offered this fittingly cryptic list, "Lights, more tassels, no unicorn, chaps, Halloween, bubble machine, battery storage, *Dumb and Dumber*."

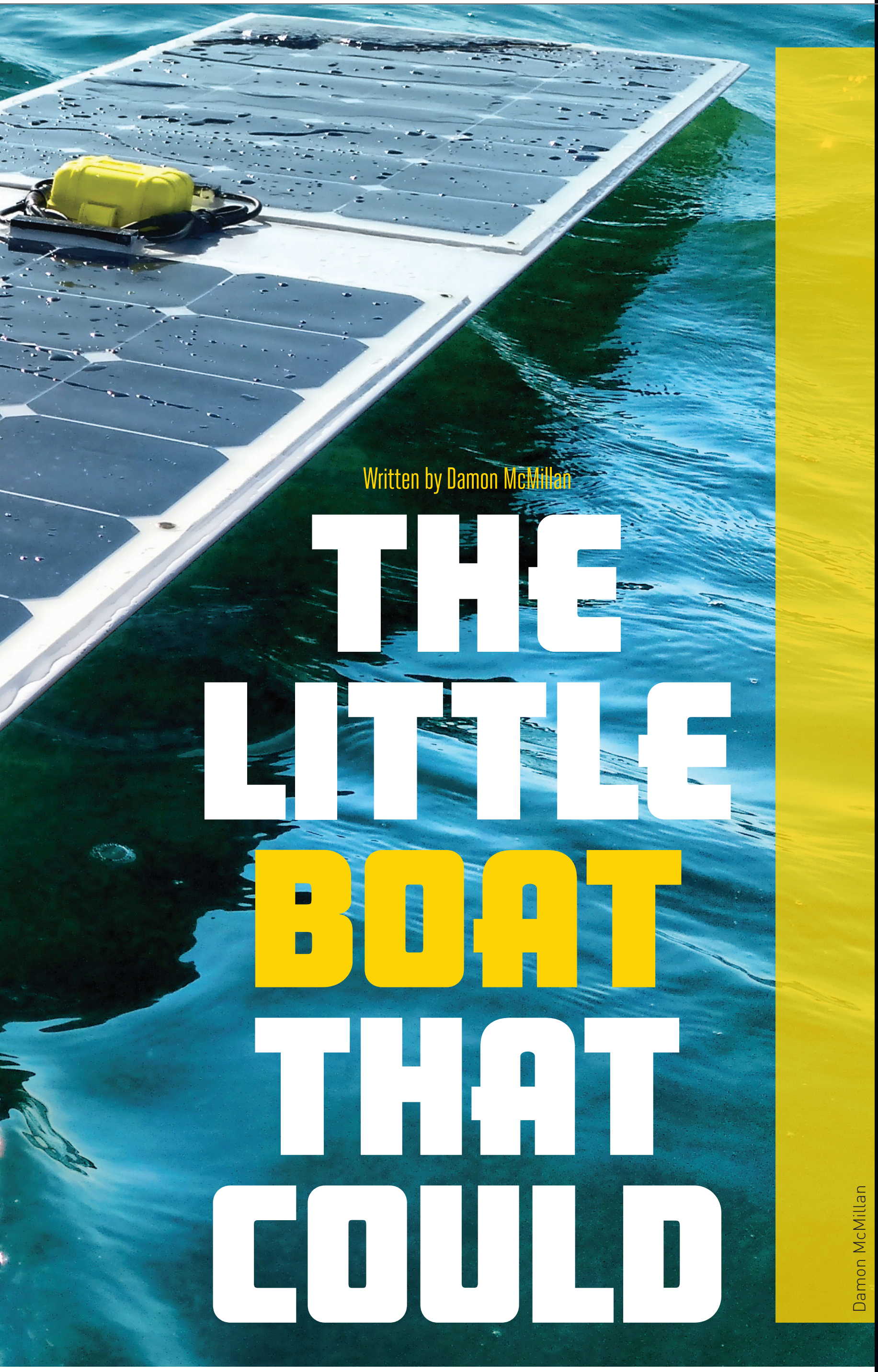
—Andrew Salomone

Bill Sitzmann



DAMON MCMILLAN is a mechanical engineer, hobbyist, husband, and father of four. He resides in Sunnyvale, California.

makezine.com/55



Written by Damon McMillan

THE LITTLE BOAT THAT COULD

Damon McMillan

THE RECORD-BREAKING TRANSOCEANIC JOURNEY OF A SOLAR-POWERED, AUTONOMOUS BOAT

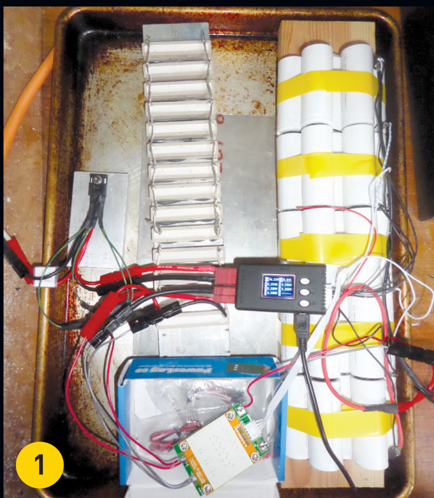
I STRUGGLE TO DESCEND THE STAIRS LEADING TO THE SANDY BEACH AT HALF MOON BAY, California, awkwardly hefting the 60-pound solar-powered *SeaCharger* atop my shoulder. Amid numerous "what-the-heck-is-that?!" stares, I perform some last-minute checks of the propeller and rudder, then wade out knee-deep and push *SeaCharger* as hard as I can toward the oncoming waves. Moving at walking speed, the boat makes it through the first several waves without flipping. Relieved, I make my way back onto the beach, then turn and watch my two-and-a-half-year project plodding westward, gradually disappearing in the whitecaps. A watchful bystander approaches me and says he's sorry I lost control of my boat and that he's sure it'll wash up on the beach. I assure him that the boat is on autopilot, going exactly where it's supposed to go. "And where is that?" "Hawaii." The look on his face is priceless.

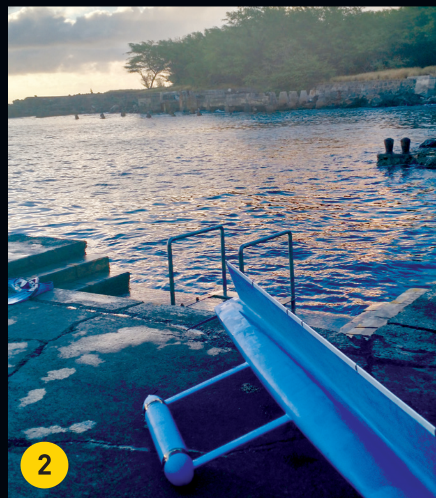
Indeed, the idea of this tiny, homemade boat surviving 2,400 miles of open ocean to reach Hawaii seems foolishly unrealistic. I know that more than anyone. With help from friends, I built the eight-foot long, foam-and-fiberglass *SeaCharger* in my garage — not for money or competition, but simply as a challenge. And a challenge it was. What started out as a yearlong project turned into 30 months of mistakes, compromises, and do-overs. So for the next few hours, I spend my time worrying and fretting, glued to my phone, waiting for each telemetry report sent by *SeaCharger*'s satellite modem. When it becomes obvious that the boat is still on track, I get in my truck and drive home.

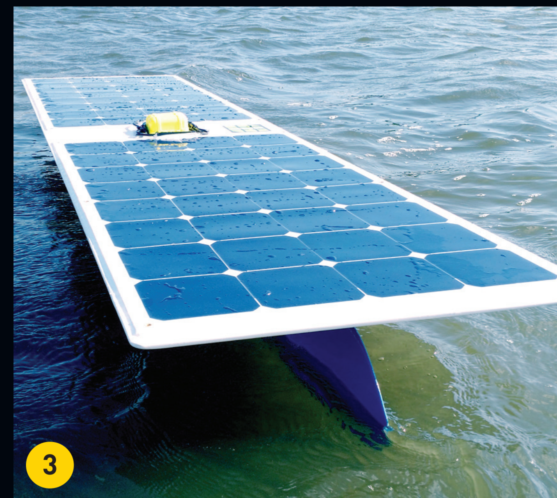
TOUCH AND GO

For the next day or two, *SeaCharger* appears to be doing remarkably well. It's windy off the California coast, and I can tell from the onboard attitude sensor that *SeaCharger* is heeling sharply. Yet the boat keeps moving west, slowly but surely, reporting in every two hours via satellite. Even at night the boat plods on with stored solar energy in its large lithium-iron phosphate battery pack. Then, after only two days, the boat fails to send its normal update. It's not completely unheard-of — sometimes satellites don't provide a strong enough signal — but it's very rare. So I anxiously wait another two hours. Again, nothing. The boat is dead, and I know it. It's never missed two updates in a row. I tell a friend that it must have sunk or been eaten by a shark. My friend says the boat will be fine. I find his reassurance unreasonable but comforting. Sure enough, two hours later, *SeaCharger* miraculously checks in again, and I breathe a sigh of relief.

Problems occur over and over again during the







1. Testing out the power system to ensure there will be no hiccups once the boat is alone on the open ocean. 2. After 41 days at sea, the *SeaCharger* takes a well-deserved break at Mahukona Harbor before the next launch. 3. The solar panels are relatively fragile, but because they are all plastic, they're almost immune to corrosive seawater.

THE ENTIRE BOAT IS AN AMALGAM OF HOBBY-GRADE/ HOMEMADE COMPONENTS AND PROFESSIONAL GRADE ONES.

next few weeks — the motor controller stops and has to be reset, strong currents almost halt the boat's progress, a cloudy day causes the boat's batteries to run out of juice. Each time, the only information I have is that the boat has stopped moving. It has no weather sensors and minimal diagnostic sensors. Extra sensors would have been expensive and risky. Increased complexity means there are more things to break.

But without weather or other information, my imagination runs wild. I often assume one of the boat's four waterproof electrical connectors has sprung a leak. I built them out of brass plumbing fittings, added O-ring grooves using a Sherline desktop mill, and potted the electrical contacts with 3M epoxy. The entire boat is an amalgam of hobby-grade/homemade components and professional-grade ones. Using purely homemade parts might jeopardize the boat's reliability, but paying for purely professional-grade parts might jeopardize my marriage. Any time *SeaCharger* stumbles, I wonder if I went cheap one time too many.

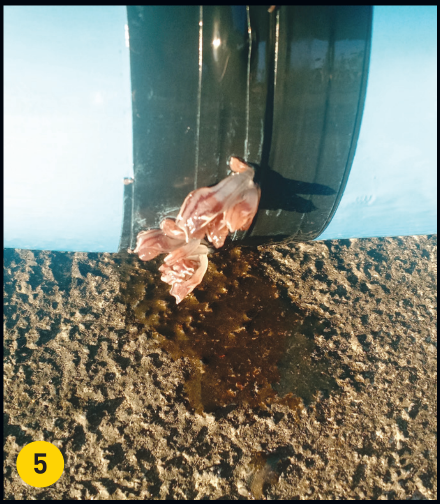
One of the biggest questions is if the boat's solar panels will survive. They're not the type you'd find on a house, as those can't be immersed in salt water. Instead, they are made only from laminated layers of plastic, without the typical aluminum and glass components found in other panels. If placed on a house, these thin, semi-flexible panels might not withstand tree branches falling on them, but there are no trees on the ocean. Extra marine sealant is applied to the panels where the electrical wires exit, and the panels are hard-wired to the boat instead of using connectors. Two solar panels are used for redundancy, but there is no redundancy in any of the other components on the boat, mostly to keep costs down.

SeaCharger uses off-the-shelf electronics as much as possible. The brains of the boat are an Arduino Mega, an Adafruit GPS, a satellite modem from Rock Seven, a compass from Devantech, and a battery protection/charging circuit from AA Portable Power Corp. A typical brushless motor spins the propeller and an R/C servo turns the rudder. I don't worry too much about the reliability of the electronics, but I do worry about the motor and servo. Water isn't the problem: the motor transfers torque to the propeller through a magnetic coupling, so it stays perfectly dry. And the servo has its own custom enclosure with rubber shaft seals to keep water out. But the bigger issue is the time required to get from California to Hawaii: the motor will have to run almost nonstop for over a month, while the rudder servo will have to complete 2 to 3 million cycles.

SMOOTH SAILING

In spite of my worrying, after 3 weeks at sea, the boat is not only still alive but is actually moving along at a very good clip. For the past two and half years, amid my children's constant cries of "Dad, when is the boat going to be done?" I've motivated myself by visualizing the moment when I'm standing somewhere in Hawaii as *SeaCharger* appears in the distance, comes motoring into the harbor, and I pull it triumphantly out of the water. Now it looks like that might actually happen!

Three more weeks pass. I'm standing on the shore at Mahukona Harbor on the Big Island with my wife, parents, brother, and a reporter from the local newspaper. I catch the first glimpse of *SeaCharger*'s solar panels flashing in the setting sun as it approaches. This moment is not as triumphant as it is surreal. This is the same *SeaCharger* that left California 41 days and 2,413 miles ago, but



Damon McMillan and Jillaire McMillan

4. My good friend JT Zemp assists with a trial launch. 5. As I suspected, the boat accumulated a number of barnacles that slowed it down. 6. I march toward the water on a beautiful day to relaunch the boat from Hapuna Beach, Hawaii, on July 27, 2016.

the faded paint and clinging barnacles only hint at what it must have experienced — and survived — to get here.

Safely ashore, *SeaCharger* appears to be in remarkably good shape. I've been asked dozens of times what I would do with *SeaCharger* after it made it to Hawaii, but I had never seriously considered it. Now it's time to decide. We could pack it in a giant crate and ship it home, but that option doesn't appeal to our pocketbook nor our sense of adventure. The truth is, the goal of this project was to send a solar-powered boat across an ocean. Making it from California to Hawaii was certainly an epic voyage, but it wasn't truly transoceanic. So I reprogram the boat to navigate across the Pacific towards New Zealand, a seemingly impossible 4,400 miles away, and launch it.

The California launch was cold, windy, and extremely nerve-wracking. But in Hawaii, there's only a minor breeze blowing over a perfect white sand beach. I wade in, waist-deep, lower *SeaCharger* into the warm water, and it casually motors out to sea. There are no nerves or tension. I feel *SeaCharger* has already proven itself.

SUPREMELY SEAWORTHY

As I write this, 90 days have passed since *SeaCharger* left Hawaii. It's crossed both the equator and the International Date Line and has traveled over one quarter of the Earth's circumference. On one or two occasions it came dangerously close to running aground on a small island, thanks to a combination of my poor planning and stormy weather. Every time *SeaCharger* nears an island, I research that island's geography and history. I learned that the Maori name for New Zealand is *Aotearoa*, or "the land of the long, white cloud." Gee, it sure would've been nice to know that before I sent my solar-powered boat there!

The Maori knew what they were talking about: there are lots of clouds around New Zealand. And as *SeaCharger* makes its approach, the wind and currents conspire against it, pushing it north as it tries to make headway south. After three months at sea, the boat is dramatically slower, likely carrying a large load of barnacles. At one point I give up on New Zealand and send a command to the boat's satellite modem to turn it west in an attempt to make it to New Caledonia or Norfolk Island. But the incredibly friendly, enthusiastic New Zealanders who are preparing for *SeaCharger's* arrival assure me that conditions will change. Sure enough, soon the winds calm and southward progress becomes possible again.

New Zealand is only 500 miles away now. I have no idea if *SeaCharger* will reach the shore alive. Whatever happens, I am both supremely pleased and completely baffled that it has come this far and lasted this long.

Perhaps the greatest surprise of the entire project has been the intense interest and unfailing support from hundreds of people all over the world. I would expect the New Zealand high school science teacher who leads an autonomous boat club at his school, and the budding engineer who grilled me on the technical details at Maker Faire, to be interested in this record-breaking boat. But it's even more surprising and satisfying to have my vintage-VW-loving brother-in-law and my wife's gardening, baking, homeschooling mom friend tracking it faithfully online. I've come to realize that this voyage has a universal appeal far beyond what I ever imagined when I began.

EPILOGUE: ADRIFT

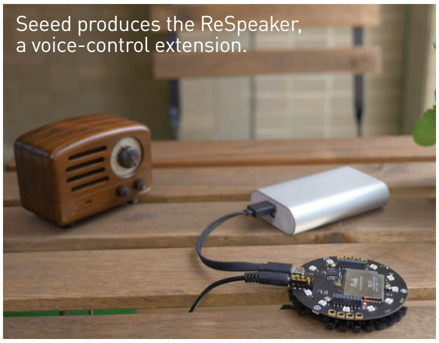
Damon reports that on November 18, after 155 days at sea, the rudder stopped responding. This time SeaCharger *really is dead, but not before it traveled an impressive 6,480 nautical miles.*

An immense thank you goes to JT Zemp, Troy Arbuckle, and Matt Stowell. Troy and Matt built the rudder actuator and helped with various electronics. JT was there from the beginning, helping with overall system architecture, fabrication, and testing.

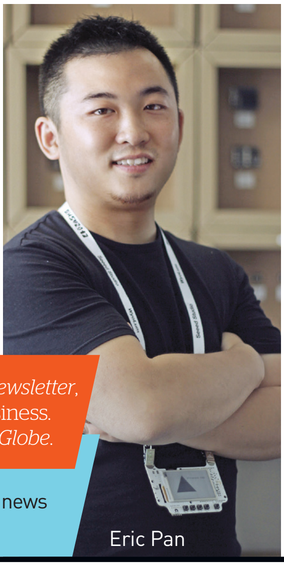
Planting the Seeed

Electronics manufacturer Eric Pan discusses the future of factories Written by DC Denison

Seeed Studio sponsors Chaihuo Maker Space, who, along with the Taiwan Robot Combat League, attracted 16 teams from all over Asia to compete at Maker Faire Shenzhen 2016.



Seeed produces the ReSpeaker, a voice-control extension.



Eric Pan

Seeed

Eric Pan started Seeed Studio in 2008, selling electronic parts from his apartment in Shenzhen. Pan also created a hackerspace in Shenzhen, co-founded the hardware accelerator HAXLR8R, and introduced the first Mini Maker Faire to China. Today Seeed has 200+ employees, and generates more than $30 million in annual revenues. Seeed's current offerings include not only parts, but also what Pan calls "0.9 kits," which aim to get buyers "almost to 1.0," while still allowing the option to change every single layer: from firmware, to circuitry, to enclosures.

Q. Are we approaching an era of "indie products," similar to indie films?

A. I think so, and it's coming from two directions. Because of the maker movement, people can better understand the possibilities and get access to the tools to create independent products. Second, the demand is more diversified than before. People don't just want mass-produced products; they're interested in more specialized items, in new niches.

Q. What are some common mistakes you see new maker pros make?

A. Many makers try to do everything, but existing businesses have too much know-how. If you want to make a new keyboard, you are competing with existing keyboard makers. It's hard to compete, but you can collaborate with them. You can work with keyboard manufacturers to invent new functionalities and designs. This is easier, of course, if you're both working with open source technologies. The goal isn't revolution, it's collaboration and evolution.

Q. How about mistakes people make when they start working with manufacturers?

A. There are a lot of surprising issues that come up at the manufacturing stage, such as with the supply chain. There's not a lot of knowledge out there about the manufacturing process. That's why we should open source not only design and layouts, but also other parts of the manufacturing process: how to test, how to do quality control, and so on. As the knowledge domains become more complete, it will get easier.

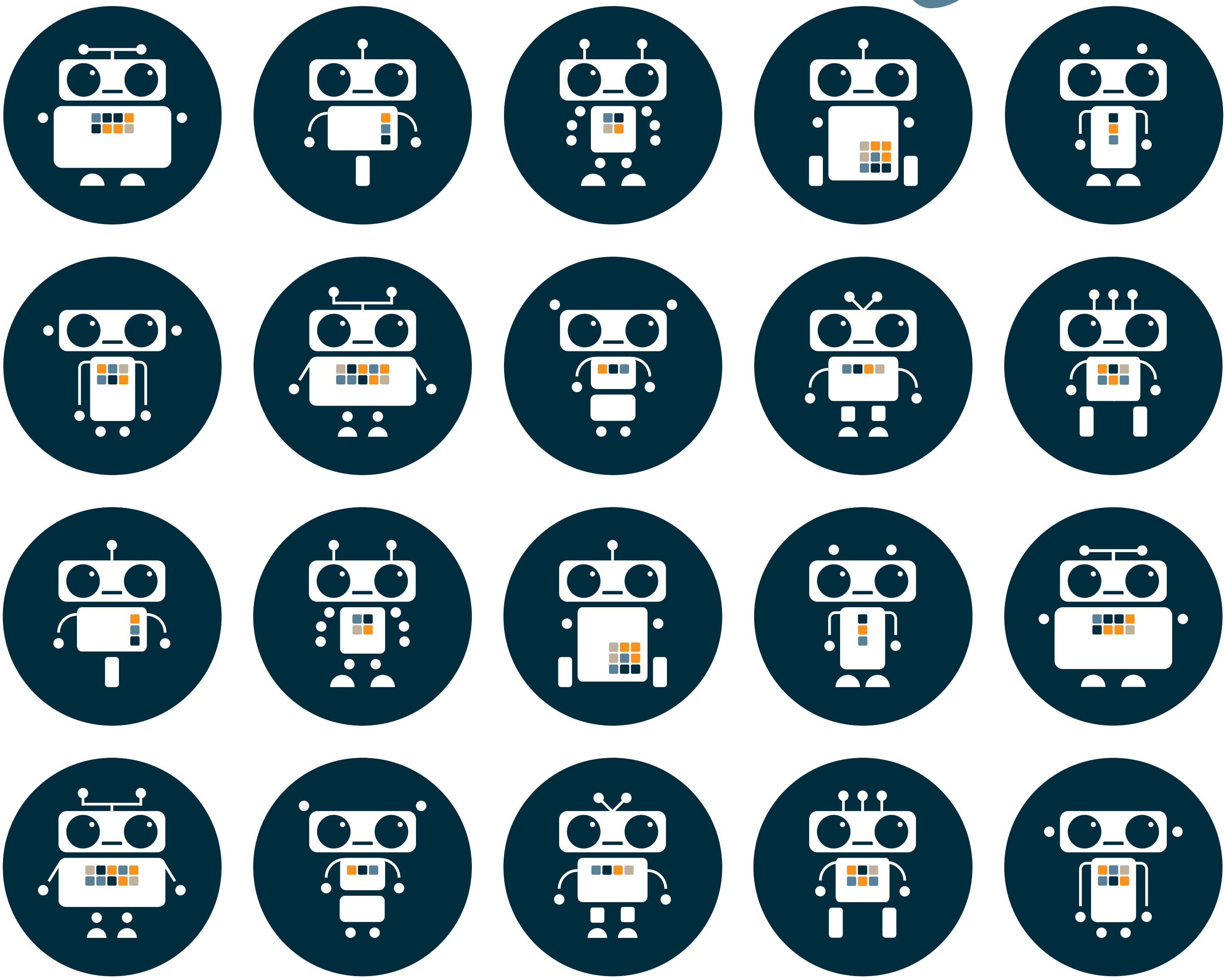
Q. Manufacturing in the U.S. versus China: has anything changed?

A. Not yet, but it's coming. More synergies in manufacturing are emerging across countries, so that it's becoming more decentralized, more like a service. In the future, more manufacturing will happen next to the consumers and clients. The future will not be bigger factories, but smaller, more agile ones.

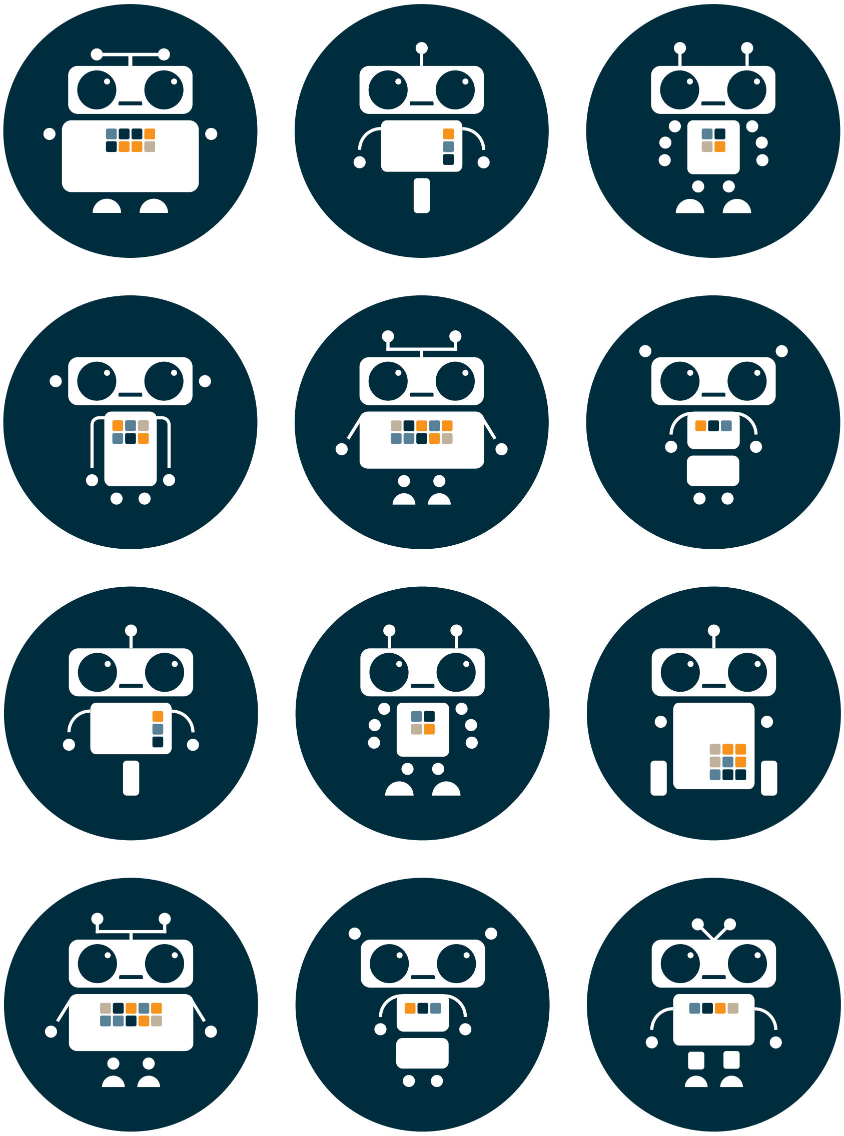
DC DENISON is the co-editor of the *Maker Pro Newsletter*, which covers the intersection of makers and business. He is the former technology editor of *The Boston Globe*.

Read the full interview, and find more Maker Pro news and interviews at makezine.com/go/makerpro.

BOT Factory



There's never been a better time to start building your own robot



Juliann Brown

WE ALL WANT TO BUILD OUR OWN ROBOT — but doing so is a daunting task that requires skill in hardware, software, and mechanics.

The good news, however, is that those who take this on will find that bot building is more accessible than ever. The motors, sensors, and hardware controllers that power these creations are easy and cheap to acquire, and offer incredible power. The software to program your machines today can be as simple as dragging blocks of code in a graphical interface (see page 33) — and even the advanced, custom options like Robot Operating System (covered on page 30) use protocols that are easily created, modified, and shared.

Hardware access has grown tremendously through the incredible adoption of Arduino and Raspberry Pi over the past decade. Beyond being solid yet simple prototyping platforms with massive communities, these movements have also introduced familiarity to a wide world of microcontrollers and single-board computers (SBCs) — the brains that control the components that drive a robotics project. The Chip-E project (page 40) uses an Arduino variant called Geekduino, and the companion balance bot EddiePlus (page 34) is controlled by the tiny, powerful Intel Edison SBC.

Both of these projects use 3D-printed frames, which is another key development that has helped further the ability to build custom robotics projects. But one of the fastest prototyping materials continues to be Lego, which has cemented itself as a key learning platform for advanced electronics builds. Check out TrotBot (page 42) and the Color-Sensing Sound Sequencer (page 44) for ways to get started using this platform.

And if you're simply looking to get a kit that contains everything you need for a certain type of robot, well, the options are now nearly boundless. We scratch the surface with our overview of kit products that are great for makers — check them out on the following page to figure out what will suit your needs best, then get started making.

Build-a-Bot

Written by Andrew Terranova

ON A ROBOT KIT QUEST? LET OUR GUIDE LEAD THE WAY

Makeblock Ultimate 2.0 10-in-1 robot kit, $500

ANDREW TERRANOVA is an engineer, maker and writer who is usually found putting something together or taking something apart. He makes robots, electronics, and other fun stuff whenever he gets the chance.

THERE ARE AN AMAZING VARIETY OF ROBOT KITS AVAILABLE, BUT WHICH ONE IS RIGHT FOR YOU? There is no such thing as the best kit, just the one most suitable for your purposes. Still, the number of choices can be confusing.

Why start with a kit? Well, there are several advantages. Generally, kits will be cheaper than buying all the same parts. The companies that put kits together buy parts in quantity, usually passing the savings to you. You also save on shipping costs, since you buy one box from one supplier. It is often a challenge to get all the parts you need for a robot from a single source, so you end up paying a significant amount in shipping costs.

Also, buying in one box means all your parts arrive at once. This saves you the annoyance of waiting for that one part on the slow boat from China that keeps you from getting started building.

Another advantage of a kit is that it is a complete design, which lets you get right into building your robot. A good kit comes with clear instructions and support.

We review several popular types of kits, share what to look for in each, and suggest examples to check out.

STARTER KITS

LOOK FOR: Good instructions and support, low price, possibility for expansion

If you are a beginner robot maker, a starter kit may be perfect for you. You want a kit that comes with good instructions and support from the manufacturer. It is easy to get discouraged, and a good kit will bring you early success and fuel your desire to learn more and go further with your new hobby. Buy from reputable companies, and check the manufacturer's website for downloadable manuals in your language, and the ability to contact them for assistance if needed. Some companies have great communities, where members help each other. If you find a kit you like and there's a member support forum, help is never far away!

Price is also a factor. You don't want to spend too much on your first robot kit. You'll undoubtedly learn more and move on to more complex robots, so your first kit should not break the bank.

Some kits serve as a base for expansion. So consider if the kit you are looking at is a dead-end, can be expanded upon, or at the least has parts you can re-use later. Yes, robot makers often cannibalize their robots to make more robots. It's a jungle out there.

The final thing to consider is programming. If you have some coding experience, you may want to lean towards a kit that is based on a programming platform you are already familiar with.

You'll find some good example starter options from Makeblock. We recommend the **mBot** kit, which is about $100. Makeblock's kits are made with high-quality aluminum and laser-cut acrylic parts. The mCore and Me Orion controller boards are 100% Arduino IDE compatible, and will also work with Makeblock's graphical programming mBlock system, based on Scratch 2.0. These boards use modular RJ25 connectors (like household phone jacks), which is nice for beginners who don't want to mess with wiring. Makeblock has parts and expansion modules so you can add more functionality to the kits.

Another great set of kits to look at is the Revolution line from EZ-Robot (see "One Million Robots," page 29). For a beginner we recommend the **Adventure Bot** due to its entry-level price of about $150. For some more advanced EZ-Robot kits, look at the **JD Humanoid** or **Roli Rover**, which seem the most popular. The EZ-Robot site also hosts an active community forum, which is very helpful.

EDUCATIONAL ROBOT KITS

LOOK FOR: Curriculum support and sturdy construction

If you are an educator, kits from companies that offer supporting curricula are ideal. You'll want durable parts that will withstand abuse class after class. A graphical programming environment is a big plus as well.

Lego Mindstorms EV3 kits are quite popular with schools for a good reason. They meet all the criteria above, including products targeted specifically for middle schools, a site dedicated for educators, and even a community website. The **Education EV3 Core Set**, which costs about $380, is a great place to start.

Vex Robotics is another popular choice with schools. Vex's RobotC programming language offers a nice graphical environment to learn on that can help students transition to C programming in the future. The solid **Programming Control Starter Kit** costs about $440.

For something a bit less traditional, take a look at the **Cubelets** kits from Modular Robotics. A 12 pack of Cubelets costs $330. This physical computing approach to robotics lets you combine different functional blocks together to make robots that can move, make sounds, light up, and sense the world around them, without even having to program them. You can expand upon the default behavior of the blocks using the graphical Blocky programming language. Like the more established Lego and Vex Robotics, Modular Robotics has support for educators, including free lesson plans.

WORKSHOP KITS

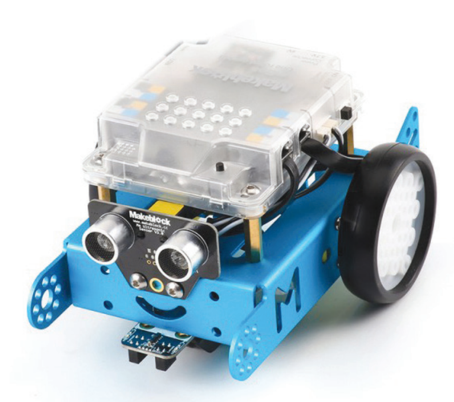
LOOK FOR: Low price, engaging design, easy to build in a limited period of time

If you've ever run a robotics workshop, you know that good instructions, simple building techniques, and the ability to complete the project in a short time frame are top priorities. The project must also be fun and engaging. We favor kits that are inexpensive enough for the participants to take home, rather than re-usable kits that must be left behind.

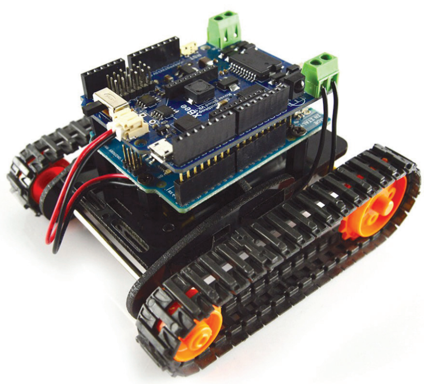
Elenco's **Line Tracking Robot Kit** is a perfect example. This cute, bright yellow, line-following kit is easy to assemble, requires no soldering, and can be completed in under an hour. We found it listed on Amazon for around $24.

For a slightly more advanced build, you can try DFRobot's **Insectbot Hexa** kit. This walking insect robot navigates autonomously with an infrared distance sensor. The Arduino-compatible board comes pre-loaded with a default sketch, but is completely programmable. It's a good value for a kit at about $38 each, and quantity discounts are available. For a short duration workshop, you can pre-build some of the more time-consuming or difficult steps and still complete the workshop in an hour.

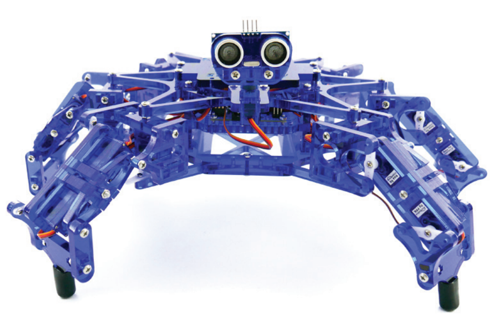
Juliann Brown, Hep Svadja



Makeblock - mBot



RobotShop - Tracked Tank Kit



ArcBotics - Hexy

EZ-Robot Revolution - Roli Rover

HUMANOID KITS

LOOK FOR: High-quality mechanical parts and servos

Humanoid robot kits can run thousands of dollars; we'll focus on more affordable options. Look for high-quality mechanical parts and servos. It should be noted that even though we humans make tasks like walking look easy, programming a biped robot can be fairly complex.

Lynxmotion makes a range of biped robot kits. The company considers their kits development platforms, meaning they generally do not provide complete code or pre-programmed apps to run your biped — you must do the software development yourself. Lynxmotion has a free Servo Sequencer utility that works with their SSC-32 servo controller. You can also add a microcontroller if you want to make an autonomous biped. They also sell a development platform called FlowBotics Studio for about $40.

Lynxmotion's kits include anodized black aluminum servo brackets, and you have options to buy the hardware alone, the hardware plus servos, or the hardware, servos, and a servo controller.

The lowest-end **Biped BRAT** has 3 degrees of freedom (DOF) for each leg, and is available as a combo kit including servos, electronics, and Lynxmotion's BotBoarduino microcontroller and example programs for about $240. The **Biped Scout** has 6 DOF per leg, and sells for about $170 without servos and electronics, which would add over $500 bought separately. These two kits are just a pair of legs and a torso. The **Biped Pete** kit, however, is a full-fledged humanoid with arms, legs, a head, and gripper hands, with a massive 22 total DOF. Pete costs $370 for just the hardware. Adding servos, electronics, and a microcontroller will cost about $550 more.

Robotis makes some very high-end biped kits, but they really hit the sweet spot in the market with their **Darwin-Mini** biped, which retails for about $500. Robotis uses its own Dynamixel brand high-quality networked servos, which are a huge step up from hobby grade servos. It works with the Robotis R+ Task and R+ Motion software, or with a smartphone app.

HEXAPOD AND QUADRUPED KITS

CONSIDER: How many DOF can you afford?

Another very popular type of kit is the hexapod walker. Things to consider are the number of DOF per leg, and the quality of the parts. Example software with pre-programmed walking gaits and maneuvers is a plus too.

A six-legged robot with 3 DOF for each leg means 18 servos. The weight of the robot and the mechanical stress on the parts means those servos and the mechanical linkages that connect them must be very high quality.

At the entry-level end, the **ArcBotics Hexy** is good for beginners at about $250. The acrylic parts are lightweight, allowing for use of smaller servos. Although the stock servos sold with Hexy have plastic gears, ArcBotics has come out with a metal gear servo that they offer to Hexy customers at a discount.

Higher end kits that use aluminum frames and servos with reliable metal or high-quality resin gears tend to be more expensive. Lynxmotion has an impressive and somewhat overwhelming choice of hexapod kits. The **AH2** kit is a 12-servo walker (2 DOF per leg) that costs about $410. Compare that to the **AH3** kit, an 18-servo walker, which costs about $940.

Fewer servos cuts weight and costs, so for an economical walking robot, consider a quadruped such as Lynxmotion's **SQ3**, which runs about $550. However, there are even lower-end kits out there yet.

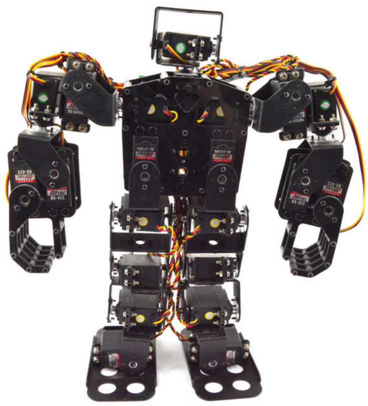
Take a look at this relative newcomer from Spierce Technologies, the **mePed v2.0 Complete Kit**, coming in under $90. Spierce had an earlier 1.0 version that was more limited, but the new kit is complete with everything you need to build it, and many design improvements over their original model.

WHEELED KITS

CONSIDER: 2WD, 4WD or more? Differential steering, or something exotic?

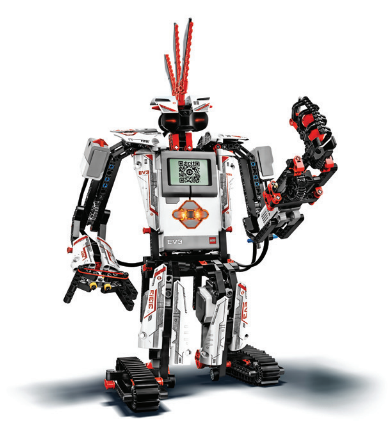
Wheeled robots that turn with the front wheels like a car are somewhat rare, as this makes navigation more complicated. Some robots can pivot the direction of their wheels like a caster to change direction, but this is also rare to find in kits. There are robots with six wheels and even special wheel designs

Makeblock, RobotShop, ArcBotics, EZ-Robot


Lynxmotion – Biped Pete


Elenco – Line Tracker


EZ-Robot Revolution – JD Humanoid


Lego – Mindstorms

(omnidirectional, Mecanum, etc.) that allow for other steering methods.

Most wheeled robot kits have either two or four wheels and use a differential drive, with powered wheels on each side that change speed and direction to turn the robot. That's our focus here.

Two-wheel drive (2WD) robots are very maneuverable, since they can spin around the center point between the two drive wheels. The Pololu **3pi Robot** is a great example and costs just under $100. The 3pi can be expanded with a second deck.

Another nice 2WD example is the DFRobot **MiniQ 2WD Complete Kit** at $80. The MiniQ also comes in a 4 wheel drive (4WD) version for about $100.

TRACKED KITS

CONSIDER: Do you need a good base for a project, or a complete, functioning robot?

Robots with tracks like a tank use differential steering like most wheeled robots. Smaller tracked robots are easy to find as kits, but larger ones are generally sold as a base chassis you can build upon to make your robot.

RobotShop makes a nice robot tank kit for about $90 with a built-in Arduino-compatible control board and an onboard LiPo battery charger. This kit can be expanded by adding an Arduino shield daughterboard, using the solder prototyping area, and with two XBee sockets for wireless communications.

Several manufacturers use Dagu's **Rover 5 Tracked Chassis** as a base, including a SparkFun kit that costs about $60. You'll need to add your own control electronics and sensors.

For a higher-end tracked robot, you could start with the Lynxmotion **Tri-Track Chassis** for around $220. Like the Rover 5 above, you'll need to add to this chassis to make a complete robot.

GET BUILDING!

The world of robotics is very broad. In addition to the categories listed, there are robotic arms, balancing robots, flying robots, swimming robots, and much more. This list should be a good starting point for the most common types of robot kits. Decide which category you are interested in, then use the examples for comparison and pick the kit that is best for you.

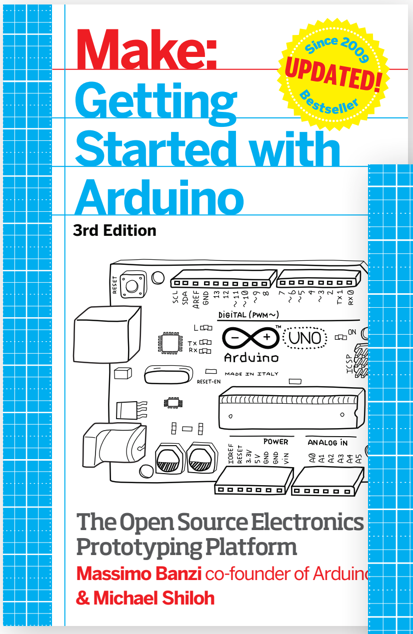
Robot Kits and Parts Makers

Looking for more robot kits? The options are numerous. This list of kit makers is still far from complete, but should cover just about any project you want to start.

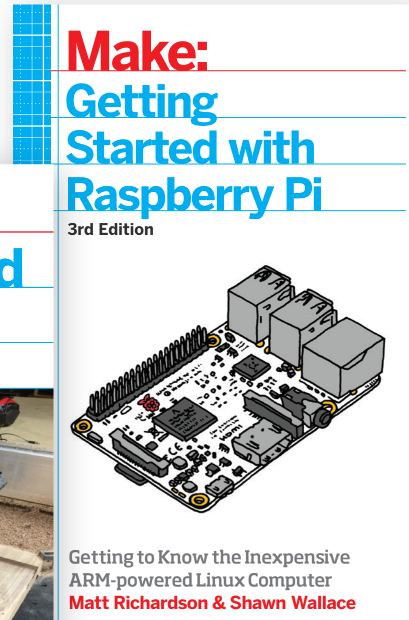
- Abilix
- Adafruit Industries
- AeroQuad
- AndyMark
- ArcBotics
- Arduino
- Artec Block
- BirdBrain Technologies
- BirdsEyeView
- Chibitronics
- Commonplace Robotics
- Cytron
- Dagu
- Dexter Industries
- Dongbu Robot
- DFRobot
- ElecFreaks
- Elenco
- EZ-Robot
- FingerTech
- Fischertechnik
- Gears EdS
- Hangfa Hydraulic Engineering
- Hexbug
- Hicat.livera
- Inspectorbots
- ITead Studio
- JCM inVentures
- Keenon Robot
- King Kong Robot
- Kondo Robot
- KumoTek
- Learning Resources
- Lego
- LinkSprite
- littleBits
- Lynxmotion
- Makeblock
- MeArm
- Microbric
- Microduino
- Mindsensors
- Modular Robotics
- Multiplo
- Nexus Robot
- OpenROV
- OWI
- Parallax
- Pittsco
- PlayMonster
- Pololu
- Quirkbot
- Revolution Education
- RoboBrothers
- RoboBuilder
- RoboCore
- Robopec
- RobotGeek
- Robotiq
- Robotis
- Robotnik
- RobotShop
- SainSmart
- Seeed Studio
- ServoCity
- SmartLab Toys
- Solarbotics
- SparkFun
- Spierce Technologies
- SunFounder
- SuperDroid Robots
- Tamiya
- Thames & Kosmos
- Tinkerbots
- Trossen Robotics
- UBTech
- Velleman
- Vex Robotics
- Wonder Workshop
- XYZ Robotics

RobotShop, Elenco, EZ-Robot, Lego

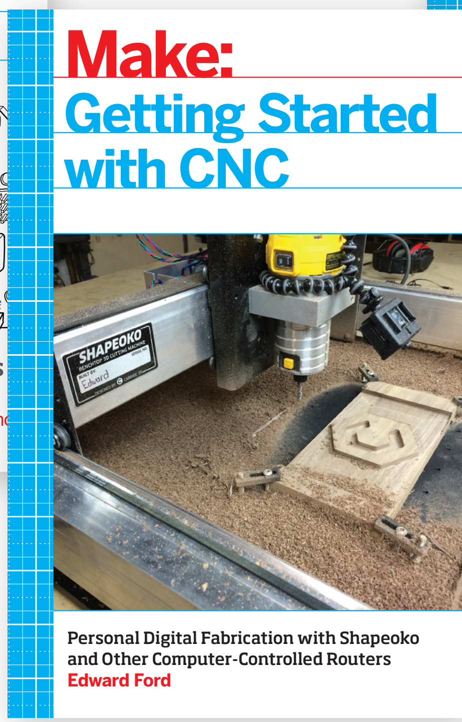
Make: books



Getting Started with Ardiuno



Getting Started with Raspberry Pi



Getting Started with CNC

Make: Books' Getting Started With series takes readers with little or no background in a subject and gently leads them step-by-step through a new technology by providing easy projects that are guaranteed to offer success.

makerfaire.com . makezine.com

My Mini Mars Rover

Written by Erica Tiberia

ERICA TIBERIA is a creative technologist, educator, and maker. She has a passion for making fearlessly to change the world.

HOW I BUILT A CENTENNIAL CHALLENGE ROBOT USING THE CHEAPEST MATERIALS I COULD FIND

Juliann Brown, Shanik Tanna

THE NASA CENTENNIAL CHALLENGES are a set of interesting contests aimed at leveraging the knowledge of citizen inventors to solve actual obstacles faced by the space agency. In June 2016, I participated in their Sample Return Robot Competition, a $1.5 million Mars-focused program aimed to create a robot that can autonomously locate, retrieve, and return various samples while navigating with limited mapping information and without the use of Earth-based navigation tools.

With a background in molecular biology, biotechnology, and research, I never expected to say that I build robots. I competed in the challenge as a single-person team against much larger universities and companies, some that had already been participating for multiple years. I based the design and programming strategy for my robot on the desire for simplicity, a small budget, and inspiration from biological systems and insects. Computer vision in variable lighting conditions and navigation without GPS or a compass were major design challenges. Preparation for the competition also involved months of field-testing and debugging. I rented out a domed soccer field to accomplish full-scale field-testing in the midst of a Canadian winter.

The Sample Return Robot Competition is a two-level challenge; Level 1 requires the return of two samples (a cylinder and a purple rock) within a 30-minute time limit. I was the first to prequalify, and the first robot on the competition field. My competition runs were less successful, but ultimately, my persistence paid off, and by the end of the week, I was one of just seven teams to complete Level 1 over the five years of competition.

Level 2 involved one single attempt on a much larger field with 11 possible samples to collect. In preparation, I built two additional robots to implement a swarm strategy that would give higher-value samples priority during transport. They worked perfectly off the field, but unfortunately the robots knocked themselves off course at the beginning of my run, ending the competition for me. Nonetheless, participating in the NASA Challenge is one of my proudest moments to date. I learned that simple solutions are often more straightforward and cheaper than those involving expensive equipment, and can be just as powerful. Also, field-testing is the most important thing you can do. Put your robot in the exact conditions that you expect to encounter, but be ready to bounce back when things don't work — because they often won't.

The competition has stuck with me, and ultimately I plan to build a company around autonomous vision-based robotics. Hopefully some of the technology and strategy from this project will make it to Mars some day.

The Robot

Other than a custom-made aluminum chassis, my robot uses common parts including a simple webcam, lidar, Dynamixel servos for the collection mechanism and camera pan and tilt, and high-torque motors for the wheels. For the brains, I used an Intel Compute Stick to run OpenCV, NumPy, and Pysolar, and an Arduino 101, which has an onboard accelerometer and gyroscope. This was the lightest and smallest robot in the competition, and I'm quite certain it also cost the least.

One Million Robots

HOW I BUILT THE EZ-ROBOT EMPIRE IN JUST FIVE YEARS

Written by DJ Sures

Juliann Brown, EZ-Robot

"YOU CAN SPEND TIME ON ONE ROBOT, OR SHARE WHAT YOU'VE LEARNED WITH THE WORLD AND HAVE A MILLION."

That's what my grandfather said to me one summer while I was struggling to build an R2-D2-style add-on for an iRobot Roomba vacuum. Without knowing complicated C, electronics, or mechanics, there were no options to build science fiction-grade robots. All we had were children's Lego toys or starting from zero.

I set out to take everything I knew about robotics and package it into the world's first complete integrated robot building platform.

My first prototype was not going to win any beauty contests, but it worked. Now all I needed was to create the software. It occurred to me that anyone building robots must already have a computer or mobile device. Wireless connectivity could enable the software to take advantage of faster processors and peripherals. Fast-forward three years, and the EZ-Builder software now hosts a mobile designer to create personalized robot apps and instantly publish to mobile devices.

Once everything seemed to work, it was time to create a circuit board and camera that people could purchase. I soldered around 100 units in my basement. They all sold in a week. My grandfather was right — I was onto something!

My talented friend Jeremie Boulianne, now EZ-Robot's lead engineer, professionally reworked my board into the popular EZ-B v3. Using the controller and software, I was able to quickly hack old toys into vision processing, speech recognizing, autonomous robots. My article "Teaching Old Toys New Tricks" in *Make:* Vol. 27 is when EZ-Robot truly took off — our forum flooded with new members and community contributions. There were new features in software updates 2 or 3 times per week. Community members were building amazing robots all over the world.

EZ-Robot raised a million dollars in angel funding and brought in an additional half a million in presales with a crowdfunding campaign to create a modular robot product called Revolution. While the bigger corporations said it couldn't be done or they weren't interested, the smaller, scrappy groups went both feet in. This helped EZ-Robot build relationships for manufacturing, assembly, quality control, and shipping logistics.

Graduating from my basement, I packed as many engineers and 3D printers that would fit into a 2,000-square-foot office. For a year we worked day and night, open sourcing the 3D-printable components, API's, and SDKs, and released an educational robot and IoT platform!

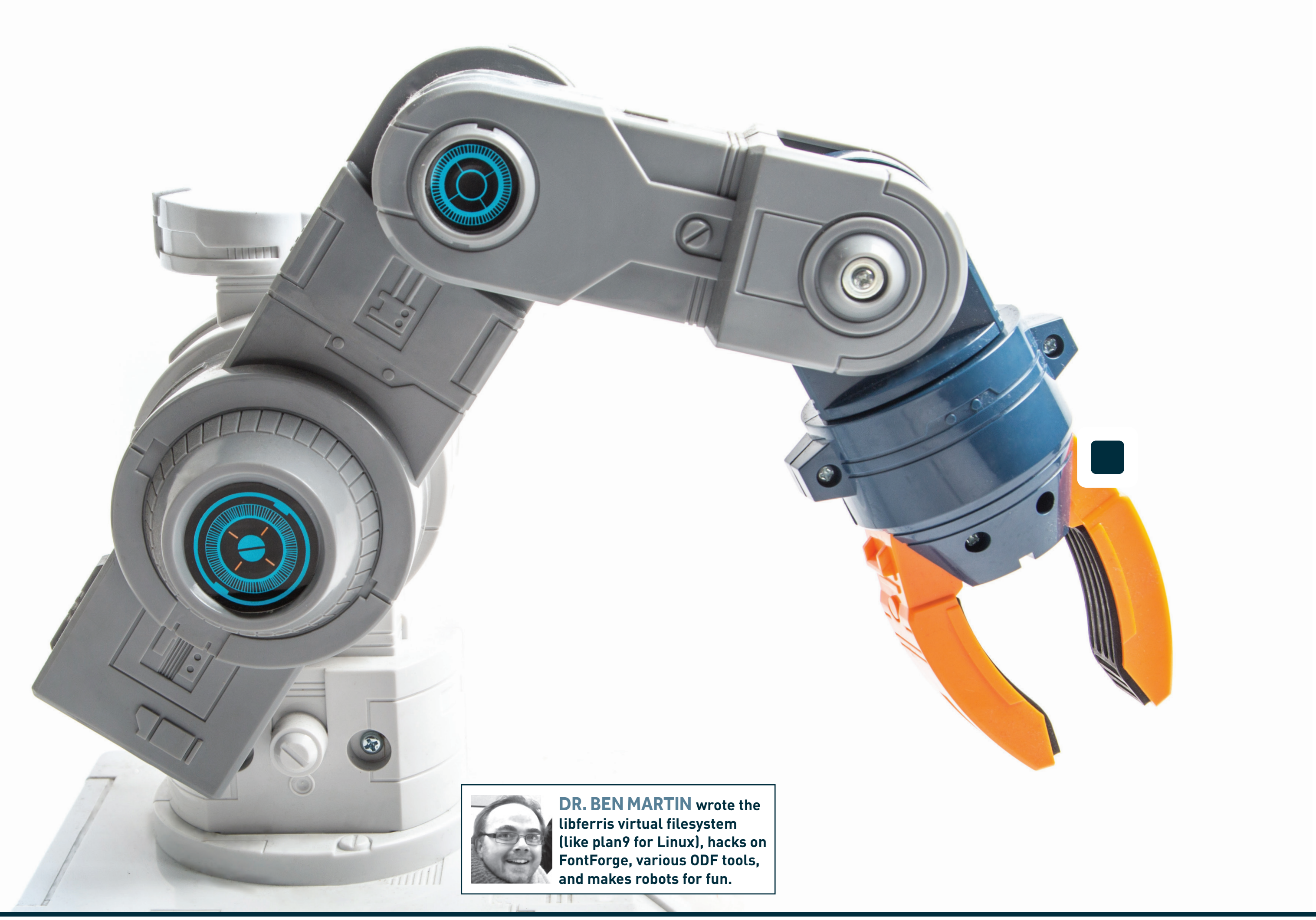
EZ-Robot now occupies a 6,000-square-foot, custom-built facility in Calgary, with a dedicated manufacturing facility in Shenzhen. There are more than 20,000 amazing robots powered by EZ-Robot. Revolution robots are actively shipping to over 100 countries. We're hard at work on the Revolution v2 platform while continuing to build relationships with similar-minded makers, geeks, and nerds who believe the real world should be a lot more like science fiction. The million-robot target is in our sights.

DJ SURES is a roboticist from Canada who began taking apart toys, stereos, and televisions at a young age. His employment history includes names like Symantec, Cisco Systems, NASA, and more.

Smooth Servo Control with ROS

Written by Ben Martin

GET STARTED WITH ROBOT OPERATING SYSTEM AND MAKE YOUR MACHINES MOVE CALMLY



DR. BEN MARTIN wrote the libferris virtual filesystem (like plan9 for Linux), hacks on FontForge, various ODF tools, and makes robots for fun.

ROBOT OPERATING SYSTEM (ROS) IS AN OPEN SOURCE ROBOTICS PLATFORM THAT HELPS YOUR ROBOT VISUALIZE THE WORLD, map and navigate it, and perform physical interactions using state-of-the-art algorithms. If you want to build a complex robot, chances are there is some ROS code already available to help you. You can use as little of ROS as you like, and it installs on machines from the Raspberry Pi level upwards.

Let's consider how to control a servo as an introduction to ROS. One drawback of servomotors is that they will often run as fast as they can to obey your command. This can result in your robot falling over because it suddenly started to rotate at top speed. Once we get ROS to control the servo, we can add sinusoidal-like control to keep your robot steady. You can do this in ROS without changing the controlling code, or the code that exposes the servo to ROS, or the servo hardware itself. And you can easily reuse the code for other projects in the future!

ROS has very good support for installation on Ubuntu or Debian, so you won't have to compile to get going. This build uses a Linux machine running Ubuntu, a hobby servo, an Arduino, and a few bits of common cables like hookup wires. (Feel free to follow along at makezine.com/go/gsw-ros, and copy the code samples there too.) ROS will be running on the Ubuntu machine and its messages will be sent over USB to the Arduino. Once you have installed the binary ROS packages, let your Arduino environment know about the ROS libraries by entering the following commands in a console program (such as gnome-terminal or konsole):

```
cd ~/sketchbook/libraries
rm -rf ros_lib
rosrun rosserial_arduino make_libraries.py .
```

PROGRAM THE ARDUINO

Now we can upload a sketch to an Arduino to perform the low-level servo control and control it from the Linux machine. This will move a servo to a location specified as a percentage (0.0 to 1.0) of the full motion we want to allow. Using a percentage instead of an explicit angle lets the Arduino code limit the exact angle that can be set, to explicitly avoid angles that you know will cause a collision.

As you can see, the normal `setup` and `loop` functions become quite sparse when using ROS. The `loop` function can be the same for any Arduino code that's just subscribing to data. In the `setup` you have to initialize ROS and then call `subscribe` for each ROS message subscriber you have. Each subscriber takes up RAM on your Arduino, so you might only have 6-12 of them depending on what else your sketch needs to do.

```
#include <Arduino.h>
#include <Servo.h>

#include <ros.h>
#include <std_msgs/Float32.h>

#define SERVOPIN 3

Servo servo;
void servo_cb( const std_msgs::Float32& msg )
{
  const float min = 45;
  const float range = 90;
  float v = msg.data;

  if( v > 1 ) v = 1;
  if( v < 0 ) v = 0;

  float angle = min + (range * v);
  servo.write(angle);
}
ros::Subscriber<std_msgs::Float32> sub( "/head/tilt", servo_cb );

ros::NodeHandle  nh;

void setup()
{
  servo.attach(SERVOPIN);

  nh.initNode();
  nh.subscribe(sub);
}

void loop()
{
  nh.spinOnce();
  delay(1); }
```

Now you need to be able to talk to the Arduino from the ROS world. The simplest way to do that is with a robot `launch` file. While the below file is very simple, these can include other `launch` files so you can eventually start a very complex robot with a single command.

```
$ cat rosservo.launch

<launch>
    <node pkg="rosserial_python" type="serial_node.py" name= "osservo" respawn="true" output="screen">
      <param name="port" value="/dev/ttyUSB0" />
    </node>
</launch>

$ roslaunch ./rosservo.lanch
```

The `rostopic` command lets you see where you can send ROS messages on your robot. As you can see below, the */head/tilt* is available from the Arduino. A message can be sent using `rostopic pub`, the `-1` option means to only publish the message once and we want to talk to */head/tilt* sending a single floating point number.

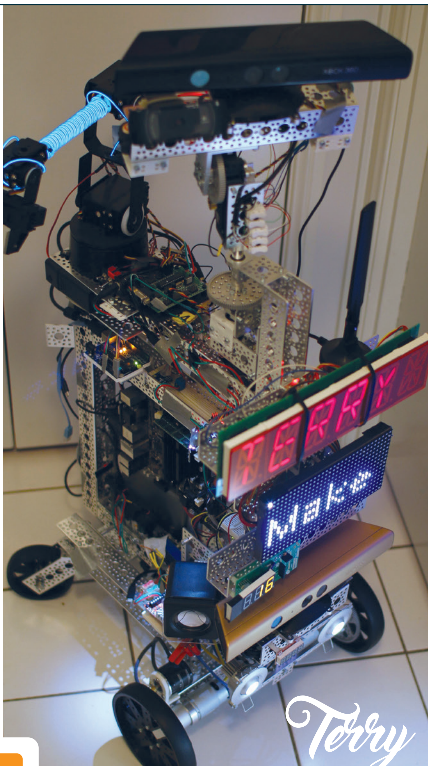
```
$ rostopic list
/diagnostics
/head/tilt
/rosout
/rosout_agg

$ rostopic pub -1 /head/tilt std_msgs/Float32 0.4
$ rostopic pub -1 /head/tilt std_msgs/Float32 0.9
```

At this stage, anything that knows how to publish a number in ROS can be used to control the servo. If we move from 0 to 1 then the servo will run at full speed, which in itself is fine, but we might like the motor to accelerate to full speed *and then slow down* when it gets near the destination position. Less sudden motion, less jerky robot movement, less surprise to the humans in the area.

SMOOTH WITH ANOTHER NODE

The below Python script listens to messages on */head/tilt/smooth* and publishes many messages to */head/tilt* to move the servo with a slow ramp up

Juliann Brown, Hep Svadja



Both Terry and the Houndbot are ROS robots made primarily out of 6061 alloy parts. My goal is to have both be as autonomous as possible.

Robot Operating System Resources

Installation on Ubuntu
wiki.ros.org/kinetic/Installation/Ubuntu

Delve into the world of navigation with ROS
wiki.ros.org/navigation

ROS Q&A
answers.ros.org/questions

Grab one of the many books on ROS
wiki.ros.org/Books

Get your robot arm on the move with ROS & MoveIt!
moveit.ros.org

Run the NASA-GM Robonaut2 in a simulator. ROS is up there!
wiki.ros.org/Robots/Robonaut2

and a ramp down when getting close to the desired position. The `moveServo_cb` is called whenever a message arrives on */head/tilt/smooth*. The callback then generates a number for every 10 degrees from -90 to +90 into the angles array. The `sin()` is taken on those angles which gives values ranging slowly from -1 to +1. Adding 1 to that makes the range 0 to +2, so a divide by 2 makes our array ramp up from 0 to +1. It's then a matter of walking through the `m` array and publishing a message each time, moving slightly further through the range `r` each time, ending up at 1*r or the full range.

```
#!/usr/bin/env python

from time import sleep
import numpy as np

import rospy
from std_msgs.msg import Float32

currentPosition = 0.5
pub = None

def moveServo_cb(data):
    global currentPosition, pub

    targetPosition = data.data
    r = targetPosition - curren
    tPosition
    angles = np.array( (range(1
    90)) [0::10]) - 90
    m = ( np.sin( angles * np.pi
    / 180. ) + 1 ) /2

   for mi in np.nditer(m):
       pos = currentPosition +
       mi*r
       print "pos: ", pos
       pub.publish(pos)
       sleep(0.05)

   currentPosition = targetPosi
   tion
    print "pos-e: ", currentPos
    ition
    pub.publish(currentPosition)

def listener():
    global pub
    rospy.init_node('servoencod
    er', anonymous=True)
    rospy.Subscriber('/head/til
    t/smooth', Float32, moveSer
    vo_cb)
    pub = rospy.Publisher('/h
    ead/tilt', Float32, queue_
    size=10)
    rospy.spin()

if __name__ == '__main__':
    listener()
```

To test out smooth servo motion, start the Python script and publish your messages to */head/tilt/smooth* and you should see a smoother movement.

```
$ ./servoencoder.py

$ rostopic pub -1 /head/tilt/
smooth  std_msgs/Float32 1
$ rostopic pub -1 /head/tilt/
smooth  std_msgs/Float32 0
```

You can also remap the name of things in ROS. This way you can remap */head/tilt/smooth* to be */head/tilt* and the program commanding the servo will not even know that the sinusoidal motion is being used.

GOING FURTHER

I've focused on simple servo control here but ROS has support for much more. If you want to know what is blocking your robot from moving, there is already support for using a Kinect in ROS. Even if the navigation stack is using that data to do mapping, you can also feed a little Python script that moves a servo to track the closest object to the robot. Yes, the eyes really are following you.

Two ROS projects of mine are Terry and Houndbot. Terry is an indoor robot with two Kinects, one used exclusively for navigation, the other for depth mapping as I see fit. With its six Arduinos, Terry can be controlled via a ROS-backed web interface or directly via PS3 remote.

I designed the Houndbot for outdoor use. It has an RC remote, GPS, compass, and ROS controlled ears. I am working on getting it to use a PS4 eye twin camera for navigation. It cannot use a Kinect because the sun stops that from working. Since the hound is about 20kg I have upgraded the suspension recently, leading me to make custom alloy parts.

Ben Martin

Programmable Bots

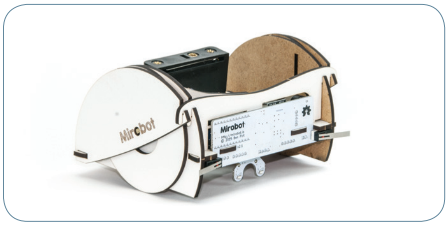
Written by Matt Stultz

LEARN TO CODE WITH THESE CLEVER NEW ROBOT SIDEKICKS

MATT STULTZ is the digital fabrication editor for *Make:* and the founder of Ocean State Maker Mill, Hack-Pittsburgh, and 3DPPVD.

Anki Cozmo

Ozobot Evo



Mime Mirobot

Julian Brown, Matt Stultz

KNOWING HOW TO PROGRAM IS AN INCREASINGLY IMPORTANT SKILL IN THE WORLD WE ARE MAKING. Thankfully, it's never been easier to learn. Block-based languages like Scratch make it simple to start acquiring the base concepts of creating software. Python, a language I personally believe should be required for graduation from high school, is now ubiquitous across almost any platform you would want to use it on. And if Arduino has taught us anything, it's that writing traditional software can be fun — but writing it to interact with the physical world is even better. In come the robots!

While Lego Mindstorms has been the undisputed champion of educational robot platforms for years, a new crop of bots are popping up that may not be as modular, but offer a lot at a fraction of the cost. I had a chance to get my hands on three of them to take their programming interfaces for a spin.

The **Ozobot Evo** packs a lot into a ping-pong ball-sized package. Evo has six programmable LEDs and proximity sensors, along with a bottom sensor for line following and color detection. The real fun comes with the included color markers for building your own mazes. The OzoBlockly programming interface allows you to decide how Evo interacts with the maze you've created, or any other challenges you want to put it through. The block programming language is a great place to get started, but hopefully the Ozo team will open up more advanced tools in the future.

The big robot buzz has been coming around the release of the **Anki Cozmo**. Cozmo is a little bot with a ton of personality. Instantly reminiscent of Disney's Wall-E, Cozmo has an LCD that makes emotive facial expressions, helping to draw you in (the built-in camera even does facial recognition — it recognizes and speaks to its user). Cozmo is loaded with sensors to discover his world, and has a forklift-like arm for interacting with it. Cozmo's API is Python-based and extensive. Still in beta, this promises to give you incredible control over Cozmo. While Python is a more advanced language, it's still very accessible to new coders and learning it will pay dividends.

My favorite is probably the least complex of the bunch, the **Mime Mirobot**. This flat-pack, laser-cut kit can be assembled without tools in a few minutes. The base function of the Mirobot is a drawbot — place a pen in its actuator and give it commands to make it draw. The power, however, comes in its array of programming options. Users can get started by programming it in multiple block languages including Scratch, but can later move on to more complicated and robust languages like Python and JavaScript. For those of us that grew up learning Logo as our first experience with programming, the Mirobot quickly becomes a real-world turtle. It works well; within a few minutes of opening their Python interface I was drawing fractals.

EddiePlus Self-Balancing Robot

Written by Renee L. Glinski

BUILD A TINY, SMOOTH-ROLLING BALANCE BOT USING INTEL'S EDISON COMPUTE MODULE — AND DRIVE IT BY FPV CAMERA

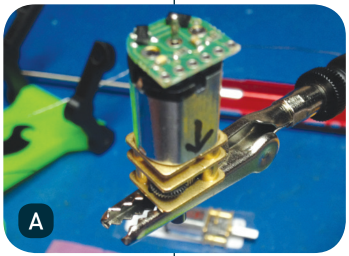
WHEN YOU'RE GETTING STARTED IN ROBOTICS, BUILDING A SELF-BALANCING BOT IS A RITE OF PASSAGE. Mine is called EddiePlus. He's rechargeable, Wi-Fi-enabled, and remote controlled, and he's an open source project, so anyone can build him in a weekend or two. He can even be navigated completely out of sight, for 1 hour of drive time, via a FPV (first-person view) camera.

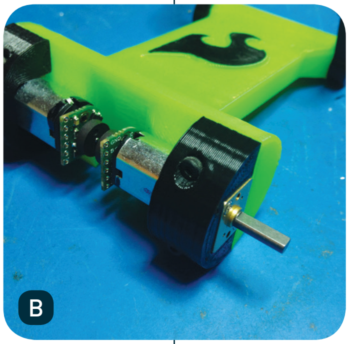
I built my first Eddie just for fun and as an excuse to put my Intel Edison Compute Module to good use. As a customer eagerly awaiting the Edison release, I was excited to have a project planned before my order arrived. The Edison's low power draw and tiny footprint make it a great fit for small, battery-powered projects that need a lot of processing power. I was also excited to try SparkFun's Edison Blocks — modular boards that stack onto the Edison to provide functions like motor control and power management.

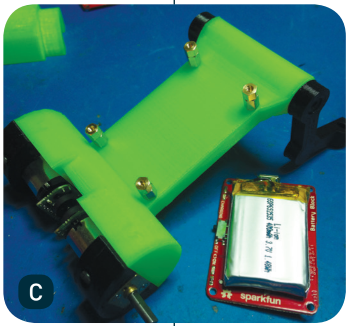
Now Eddie is back and he's excited to show off his improvements. EddiePlus has a cleaner look and vastly improved performance. He's got all the original features, plus new motor encoders, dynamic balance, dual PID control on each wheel, and faster motors and bigger wheels for more speed. His design has been updated to add interchangeable heads and a "backpack" enclosure to keep the electronics safe.

Eddie has Windows, iOS, and Python apps you can use to command him via Wi-Fi. His source code is shared at github.com/r3n33/EddieBalance, and I'm maintaining an assembly guide on Google Docs (makezine.com/go/eddie-balance).

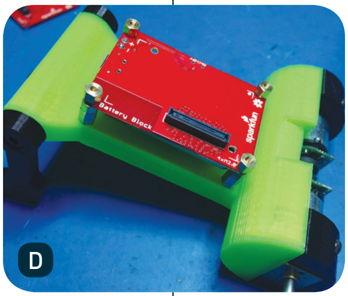
Let's build him!


A


B



C



D

1. INSTALL THE ENCODERS

Notice the arrow that I drew on the motor? It's pointing toward the side of the gearbox that has no gears in the first half (Figure A). Orient the encoder's 6 connectors toward this side of the motor as shown, then follow the manufacturer's instructions at pololu.com/product/2598 to solder them in place and install their magnetic discs.

Note Be sure to install the encoders nice and flush with the bottom of the motor.

2. MOUNT THE MOTORS

Now install the motors on the EddiePlus body using the printed motor mounts and M2×14mm hex screws or self-tapping screws. Orient the motors with their encoders as shown in Figure B, and press-fit them into the indentations inside the motor mount. Once installed, the motors should be almost perfectly flush with the edge of the body and motor mount.

Note These photos were taken when I was using M2 screws to hold the motor mounts. After stripping out too many Eddie bodies, I changed the motor mount design to accommodate more robust self-tapping screws. As a bonus, the screw heads are completely hidden in the new motor mounts.

3. INSTALL THE BATTERY BLOCK

First attach the M2×5mm standoffs to the back of the EddiePlus body, followed by a set of M2×3mm standoffs (Figure C) to give the Battery Block enough clearance (Figure D). Also make sure that the battery fits within the standoffs; you'll probably need to reposition it.

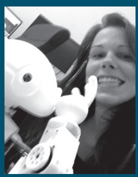
TIME REQUIRED: 1–2 Weekends
COST: $250–$325

MATERIALS

- **Intel Edison single-board computer** SparkFun #13024, sparkfun.com. Or get additional breakout boards for experimenting, with our Intel Edison Breakout Board Kit or Intel Edison Kit for Arduino, from makershed.com.
- **3D-printed robot parts: body (1), head (1), motor mounts (2), encoder cover (1), backpack (1), and arms (2)** Download the free 3D files from thingiverse.com/thing:694969. The 2 arms are in a single 3D file. There are 2 different heads you can try, or you can substitute a head of your own choosing.
- **Micro gearmotors, metal, 75:1 ratio (2)** Pololu #2215, pololu.com
- **Magnetic encoders (2)** sold by the pair, Pololu #2598
- **Wheels, 70mm × 8mm (2)** sold by the pair, Pololu #1425
- **SparkFun Blocks for Intel Edison: Dual H-Bridge motor controller (1), 9 Degrees of Freedom IMU (1), Base Block (1), GPIO Block (1), and Battery Block (1)** SparkFun #13043, 13033, 13045, 13038, and 13037
- **Ribbon cable, 6 wires wide, about 220mm total length**
- **Resistors: 5kΩ (4), and 3.9kΩ or 1.8kΩ (1)**
- **USB mini webcam (optional)** I found a low-resolution cam for $5 on eBay that clamps right onto Eddie; Amazon #B00UNIU1E4 looks even better.
- **Micro-USB cable (optional)** You'll solder it to the webcam's PCB.
- **Machine screws, spacers, and standoffs** Visit the project page at makezine.com/go/eddie-plus-fpv-balance-bot for a complete listing. You can use the little hardware packs from SparkFun (#13187) or find equivalent Uxcell parts on Amazon.

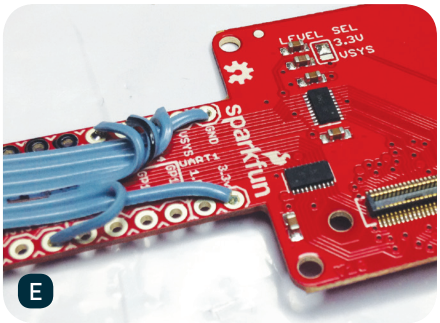
TOOLS

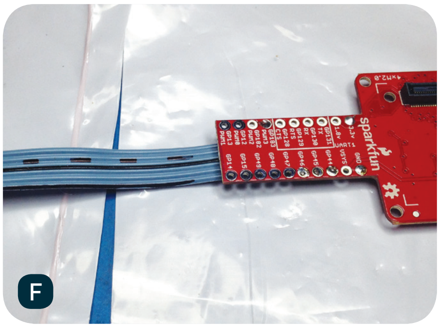
- **Soldering iron and solder**
- **Computer with internet connection and Wi-Fi network**
- **Wire cutters and strippers**
- **Razor or hobby knife**
- **Sharpie**
- **Electrical tape**
- **3D printer (optional)** To find a 3D printer or a printing service you can use, check out makezine.com/where-to-get-digital-fabrication-tool-access.

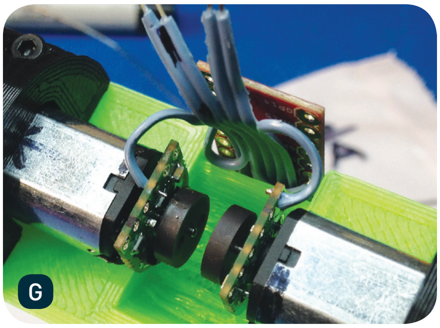


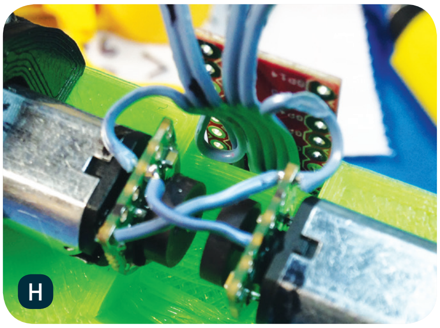
RENEE L. GLINSKI loves bringing life to robots of all kinds! Robotics, engineering, prototyping, and programming are just a few of her obsessions. Follow her at youtube.com/LoveRoboticsEngineering or instagram.com/roboempress.

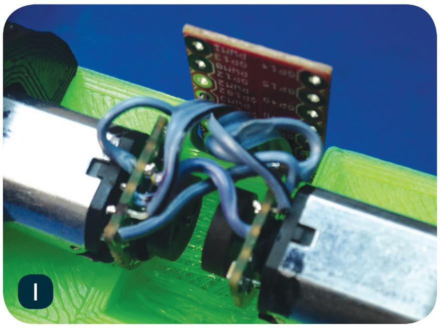
Juliann Brown, Hep Svadja, Renee L. Glinski

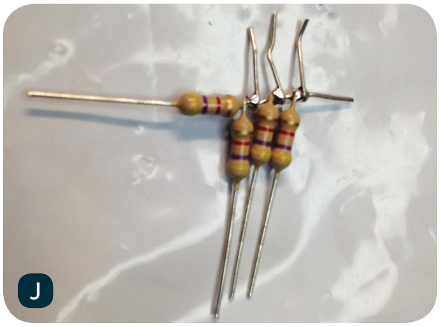


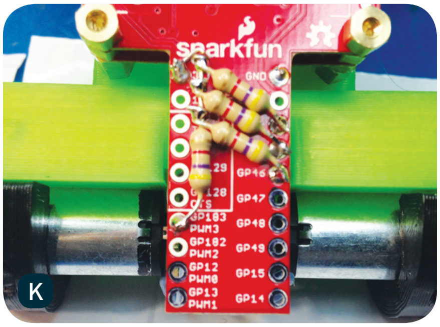


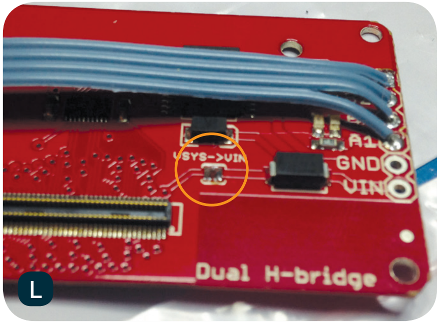


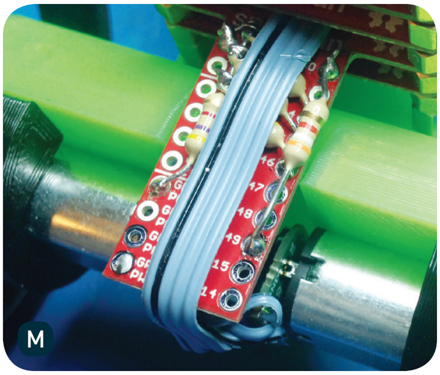


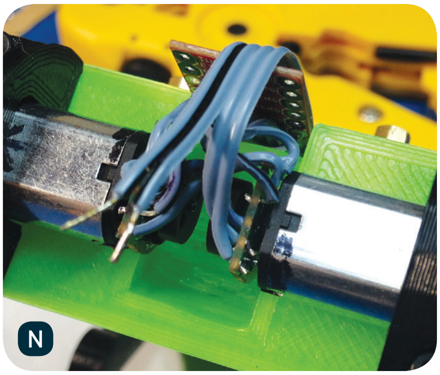












Note Every Battery Block I've received has had the battery pre-installed, requiring me to unstick the battery so it can be repositioned to fit within the 4 standoffs. Take care not to bend the PCB if you have to re-stick your battery.

4. INSTALL THE GPIO BLOCK

This is where the fun begins! You'll need a short section (about 100mm) of ribbon cable 6 wires wide. Prepare one end of the cable by stripping the shielding on all 6 wires and tinning the ends (applying solder) to keep them from fraying apart.

Start by soldering the leftmost wire of the ribbon cable (as viewed in Figure E) to power on the GPIO Block, then solder the rightmost wire to GND on the opposite side.

Using a razor or hobby knife, carefully separate the middle 4 wires to reach the GPIO breakouts, then solder them, from left to right, to GPIO pads 44, 45, 46, and 183.

Note I'm using 3.3V levels but if you're so inclined you could use VSYS (~4VDC).

Using a Sharpie I marked the edge wire for GND and added dashes to the 2 wires used on the left encoder (Figure F). I also blackened the seven GPIOs that are not available due to the Dual H-Bridge Block.

5. WIRE GPIO BLOCK TO ENCODERS

Yes, the soldering fun continues. First, stack the GPIO Block on the Battery Block with another set of 3mm standoffs.

Solder the 3.3V wire to one encoder's VCC pad, and the GND wire to the other encoder's GND pad (Figure G). Then, using 2 short wires, bridge both power and ground to the opposite encoder (Figure H).

Finally, solder the encoders' signal wires to pads A and B on both the left and right encoders (Figure I). Here's the connection table:

ENCODER TO GPIO CONNECTION

GPIO 44	RIGHT ENCODER	B CONNECTION
GPIO 45	RIGHT ENCODER	A CONNECTION
GPIO 46	LEFT ENCODER	B CONNECTION
GPIO 183	LEFT ENCODER	A CONNECTION

Once you've made it this far, go ahead and give yourself a pat on the back. Be sure none of the wires are touching the encoder magnets, and you should be in good shape.

Renee L. Glinski

Now, because the GPIO Block uses the TXB0108 chip for level shifting, you need to add a 5K pull-up resistor to each of the 4 encoder signal connections (Figures J and K). Without these resistors the auto-direction sensing in the TXB0108 will not allow the encoder signals to reach the Edison.

Note These are the smallest 5K resistors I had on hand. They work but are a little bulky.

To prevent motors from running during boot, a pull-down resistor is also used on GPIO 49. This prevents the TXB0108's default high-impedance state from bringing the dual H-bridge out of standby mode while the system is booting. I tested these:

TESTED PULL-DOWN RESISTORS ON GPIO 49

4.7K	NO	STAYS HIGH AT BOOT
3.9K	YES	WORKS
1.8K	YES	WORKS

6. WIRE AND INSTALL THE DUAL H-BRIDGE BLOCK

Start with about 120mm of ribbon cable that's 4 wires wide. Solder these to pads A1, A2, B1, and B2 on the Dual H-Bridge Block.

Next add a solder bridge to connect the VSYS->VIN pads (Figure L). This will allow the battery power to drive the motors.

Install the PCB using 3mm standoffs and wrap the wires around to the bottom side (Figure M). Be sure not to pinch or put a kink anywhere in the ribbon cable, as this will diminish the motor's drive power.

The wires on the right side go to the right motor (Figure N). The rightmost wire is B1 on the H-bridge and connects to M1 on the encoder. The adjacent wire connects to M2 on the right encoder.

The remaining wires on the left side go to the left motor (Figure O). The leftmost wire is A1 on the H-bridge and connects to M1 on the encoder. The adjacent wire connects to M2 on the left encoder.

7. INSTALL THE BASE BLOCK

Before you install the Base Block, I recommend putting about 3 layers of electrical tape over the USB connectors so no conductive metal is exposed (Figure P). If you don't, you'll see that they're dangerously close to contacting the H-bridge output.

8. INSTALL THE IMU AND THE EDISON

The 9DOF IMU is the last block in the stack and you'll secure your Edison on top of it (Figure Q).

On this build, I used the M1.5 screws and M1.5×3mm spacers from an Edison Mini Breakout Kit. I've also had success with M2×3mm screws and standoffs, but watch out for the tiny traces next to the holes. You don't want to cause any trouble.

9. INSTALL MOTOR COVER AND WHEELS

The motor cover should press-fit with a minimal amount of pressure and stay in place without screws (Figure R). There's enough clearance for the ribbon cable to wrap around the underside of the GPIO Block and not be pinched by the cover.

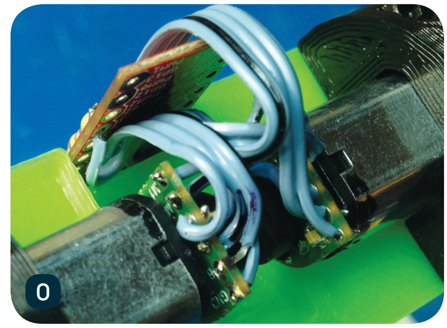
When installing the wheels I recommend using your index finger and thumb to apply pressure to the motor mount and body while pushing the wheel in place. The motors do lock in place when installed properly but I'd hate for something to go wrong when pushing the wheel onto the output shaft.

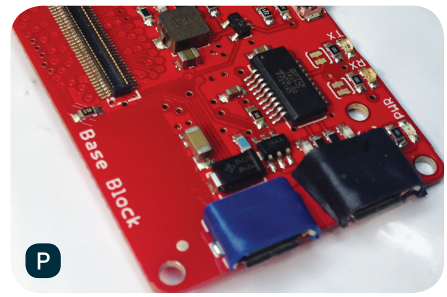
10. FINISHING TOUCHES

Slip the "backpack" cover over the electronics stack, then install the arms using the M3×14mm screws (Figure S). You can adjust their position so that Eddie can carry a variety of loads.

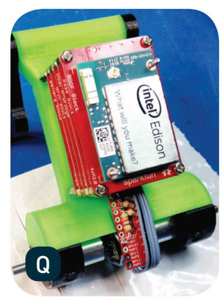
Finally, install the head of your choice using the M3×5mm screw (Figures T and U) — or mount your USB webcam instead (Figure V).

Congratulations! You've made it. You're ready to install the EddieBalance code and bring your robot to life.


O



P



Q


R


S


T


U


V

Setting Up Your Edison for Eddie the Balance Bot

» UPDATE THE OS

Download the latest Yocto image from software.intel.com/en-us/iot/hardware/edison/downloads. Then follow the Intel Flash Tool instructions for your operating system at software.intel.com/en-us/articles/flash-tool-lite-user-manual.

» LOG IN VIA SERIAL

Now you need to establish a serial connection to Eddie. To do this, connect a micro-USB cable to your computer and to Eddie's right rear USB port (the one labeled "Console"). Turn him on and place him lying on his backpack.

WINDOWS: Use Putty to establish a serial connection with the USB Port under the Serial tab in the Putty config screen, and make sure the Speed value is set to 115200 baud for Eddie's communication. If you're unfamiliar with Putty, just Google "set up putty serial".

OS X: Plug the USB cable into your computer, open Terminal and type:

```
ls /dev/ | grep usb
```

Copy the **`tty.usbserial-[SOMEVALUE]`** to a text file. Using this value, type"

```
screen /dev/tty.usbserial[SOMEVALUE] 115200
```

to connect to Eddie.

Once the Edison is booted, press Enter in the terminal and it will prompt you to log in. The login is **`root`**; there's no password yet.

» CONNECT EDDIE TO WI-FI

Now that you're logged into the Edison, run:

```
configure_edison --setup
```

Follow the prompts to configure your hostname and password, and to connect Eddie to your Wi-Fi network.

> **Note** If you ever need to connect Eddie to a different Wi-Fi network, log in and run:
>
> `configure_edison --wifi`
>
> to set up a new connection. Reboot, and Eddie will be discoverable by his apps if you've configured the startup service in the *extras* directory.

» INSTALL DEPENDENCIES

Yocto Linux uses opkg as its package manager. AlexT's unofficial opkg repository is highly recommended for adding packages to your Edison. It includes many useful packages such as **`git`** and **`libmraa`**. To configure the repository, first open the file */etc/opkg/base-feeds.conf* and add the following lines:

```
src/gz all http://repo.opkg.net/edison/repo/all
src/gz edison http://repo.opkg.net/edison/repo/edison
src/gz core2-32 http://repo.opkg.net/edison/repo/core2-32
```

Then, still logged into the Edison, run:

```
opkg update
opkg install git
```

» INSTALL THE EDDIEBALANCE SOFTWARE

Recommended from */home/root*, now run:

```
git clone https://github.com/r3n33/EddieBalance.git
```

To ensure everything is ready to go, run:

```
/home/root/EddieBalance/src/./build
```

No output means everything compiled OK and you're ready to test your Eddie bot.

To test your Eddie robot, now run:

```
/home/root/EddieBalance/src/./EddieBalance
```

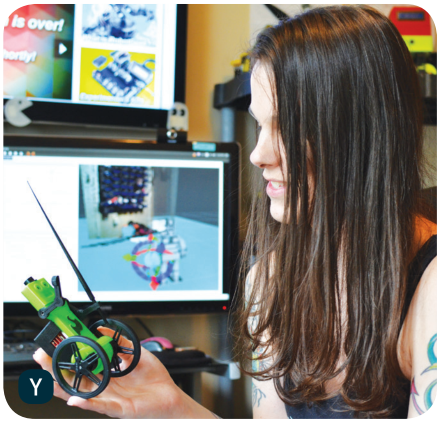
If you'd like to make the EddieBalance software run at boot, follow the instructions found in the readme file at */home/root/EddieBalance/extras/README.md*.

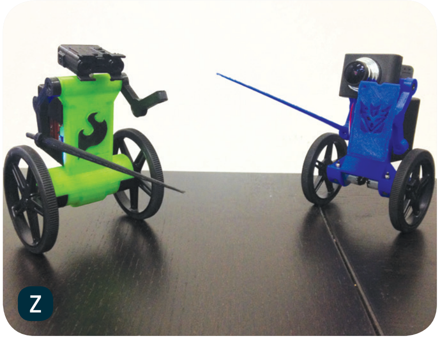
The EddieBalance software listens for data on 2 ports; one for gaining control and the other for sending commands. And Eddie returns data to the last IP received on the response port.

Follow the instructions on the Github page at github.com/r3n33/EddieBalance to "bind" Eddie to your computer for remote control. The bind process was a necessary evil to allow multiple Eddie robots on the same Wi-Fi network. Bluetooth control is next on my list of features.


W


X


Y


Z

Renee L. Glinski

LET'S ROLL!

DRIVING YOUR EDDIE BALANCE BOT

EddiePlus loves to dance, can adapt to varying carry loads (Figure W), and doesn't mind slopes (Figure X). I made some lances and shields for my Eddies (Figure Y) and then pitted them against each other in a jousting tournament (Figure Z).

You can drive your Eddie via remote control over Wi-Fi, using one of 3 apps:

» I've shared a **Windows application** for controlling Eddie, that lets you view and manage the PID data (Figure AA) and even drive around using WSAD keys. Download it from my server at labrats.io/EddieUDP.exe.

Place Eddie on the floor upright, power him up, and gently hold him until he self-balances, being careful of his backward movement. Connect your computer to the same Wi-Fi network you connected Eddie to, and run the control software. At the bottom of the panel, click the Find button and choose your Eddie's profile from the list. Near the top right of the panel, press Stream On. You should start receiving telemetry data and you can now control Eddie's motion with the WASD keys.

» I've also developed an **iPhone app** (Figures BB and CC) for controlling Eddie; it's a beta but you can install it using Impactor following the instructions at github.com/r3n33/EddieRemoteiOS.

» There's also a **Python-based controller** written by Eddie builder William Radigan, at github.com/WRadigan/PyEddieControl.

ADDING AN FPV CAMERA

I have done full FPV remote control of Eddie by streaming the video to a web browser, using edi-cam software (github.com/drejkim/edi-cam). I've also successfully streamed video to my iPhone app, but there hasn't been a lot of development time for that and it's currently disabled in the app (maybe you can help develop it!).

The main problem I had was the video had a "bobbing" effect due to the camera being fixed on the pitch axis. Still, I was able to drive Eddie around my (former) office while he was out of sight. Definitely fun! Then again if a robot is involved I'm almost automatically having fun.

The USB camera I used was a $5 eBay item. Find the smallest one you can, then it's a simple matter of soldering a short micro-USB cable to the camera's PCB, to plug into the OTG port in the Base Block.

GOING FURTHER

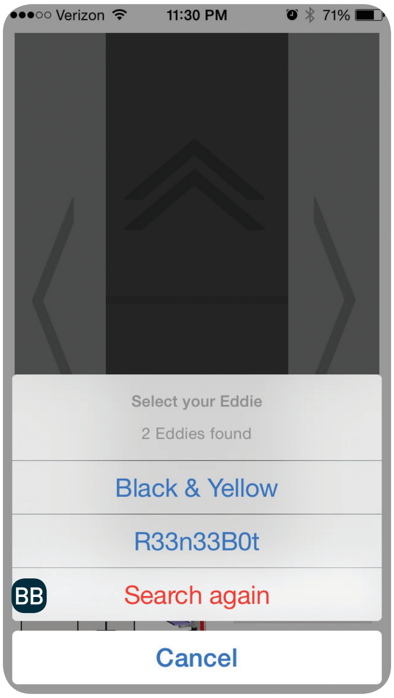
Sure, there are easier and cheaper ways to build a balancing bot, but they'd come at a cost — like size, features, or weight. For the feature set Eddie has, I don't think you can do much better on the price unless you're a very savvy shopper. You can sometimes find SparkFun coupons, or open-box Edisons for half price at Micro Center. If I were to rework this project to cut costs, I'd eliminate the SparkFun Blocks and design my own PCB with the required components.

It's easy to make this project your own by editing the 3D-printed parts to give Eddie different looks. For starters, you can model alternate heads (Figure DD) and allow room for a hex standoff in the base to make them interchangeable.

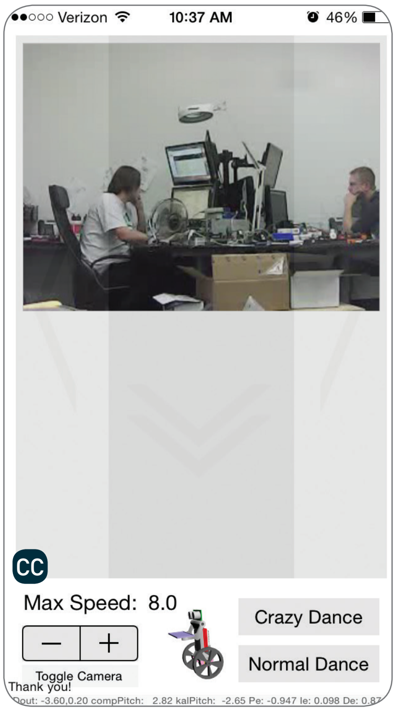
I'm also working on Bluetooth control, but haven't finished yet; I've shared my updates and findings at a git branch at github.com/r3n33/EddieBalance/tree/EddieBluetooth.

I hope you enjoy your Eddie, and I hope you'll expand and improve upon my design. If you do — or you just need help building your Eddie — please send me a message on Twitter @EddieBalance.

AA



BB



CC

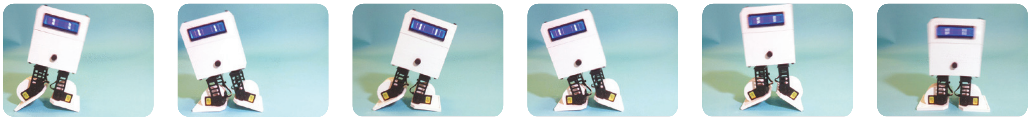
DD

Watch Eddie in action and share your build at makezine.com/go/eddie-plus-fpv-balance-bot.

Chip-E Bipedal Robot

Written by Renee L. Glinski

BUILD AN ARDUINO-POWERED, 3D-PRINTED, MOON-WALKING BOT WITH PREPROGRAMMED AND CUSTOM GAITS



BIPEDAL ROBOTS AREN'T ALWAYS EASY TO BUILD (THIS ONE IS) BUT THEY'RE SO HUMAN! I built my first humanoid robot when I was 8. It rolled on training wheels, with pseudo-circuits built from HVAC foil tape and old cassette deck switches. A now-infamous robot claw pincher arm shook people's hands — and sometimes gave them a mild shock from a step-up transformer inside. Thankfully, I did get nicer as I grew up.

As a DIY robotics hobbyist I developed open source robots like Eddie the balance bot (see page 34) and made contributions to some of Trossen Robotics' more advanced humanoids, the HR-OS1 and HR-OS5. Now I've begun working with Trossen and its subsidiaries Interbotix Labs and RobotGeek, doing what I love: making robots do amazing things. One of my newest designs is a robot that I think is quite adorable: Chip-E.

TIME REQUIRED:
A Weekend

COST:
$150–$200

MATERIALS

- **3D-printed robot parts (1 each): Base, midsection, head, top plate, right foot, left foot, right shoe, left shoe** Download the free 3D files from thingiverse.com/thing:1795648 and print them out, or have a service print them for you, such as Shapeways or 3D Hubs.
- **RobotGeek Chip-E Kit $150** from robotgeek.com/robotgeek-chip-e.aspx, includes the following parts:
  - **RobotGeek Geekduino microcontroller board** #RG-GEEKDUINO
  - **RobotGeek Sensor Shield** #RG-SENSHIELDV2
  - **RobotGeek IR Receiver** #ASM-RG-IRRECEIVER
  - **RobotGeek Buzzer** #ASM-RG-BUZZER
  - **LCD display, 2×16,** serial #SS-LCD23154P
  - **Servomotors, 6V, 180° (4)** #ASM-RGS-13
  - **C brackets, 50mm × 25mm × 55mm (2)** #RG-TR00035
  - **Battery, lithium ion, 7.4V 2200mAh, with charger** #BAT-LION7V2200 and CHG-LION
  - **Universal IR game-pad** #UG-IRGAME
  - **Bolts, nuts, standoffs, and cables** For the complete listing see learn.robotgeek.com/projects/297-chip-e-assembly-guide.html.

TOOLS

- **Computer with internet connection and Arduino IDE software** free download from arduino.cc/downloads
- **Hex drivers: 2.5mm and 1.5mm**
- **Pliers**
- **3D printer (optional)** To find a machine or a printing service you can use, check out makezine.com/where-to-get-digital-fabrication-tool-access.

RENEE L. GLINSKI loves bringing life to robots of all kinds! Robotics, engineering, prototyping, and programming are just a few of her obsessions; follow her at youtube.com/LoveRoboticsEngineering or instagram.com/roboempress.

Julian Brown, Hep Svadja, Renee L. Glinski

Chip-E started as a personal project to cheer me up during a rough time due to the loss of a family member. Having a supply of RobotGeek components on hand, I thought about a simple bipedal design, but having built some of the other 4-servo bipeds out there, I wanted to do something different.

Chip-E has an LCD for "eyes" enabling visual feedback, a piezoelectric buzzer for auditory feedback, and an IR sensor for control. These features are complemented by 4 strong RGS-13 servos and the ability to add more sensors. Like his sibling Eddie, Chip-E is designed to be easy to 3D print and to bring joy to those who appreciate him. (Many makers may recognize the symbolism of the "dot" and notched corner, giving him the look of a microchip.)

BOTS MADE FOR WALKING

Bipedal robots come in many configurations. Some are simplified and focused on efficiency while others can be complex and challenging to balance. The movement pattern of the legs to produce locomotion is known as the *gait*. In a robot like Chip-E, gait generation is simplified to an oscillation of the servo positions to produce different patterns of movement. Chip-E's large footprint and high servo torque provide definite balancing advantages to this design.

We at Trossen have made the RobotGeek Chip-E robot our first 3D-printable kit: You print the body and we provide all the components in a handy kit. Included is our new universal IR game-pad for wireless control, with lots of buttons for performing actions or changing settings; its dual mode even lets you control 2 robots. Chip-E's code is based on the Zowi project and is compatible with BoB, Otto, and other open source bipeds.

Having 2 legs means Chip-E can walk, dance, or even shuffle his way around your workspace or play area. And because he's small and tough, he's ready to come along on an adventure and see the world.

BUILDING YOUR CHIP-E

Complete instructions are available at learn.robotgeek.com/projects. Here's an overview.

1. SET UP THE GEEKDUINO

Download Chip-E's code and the necessary libraries to your computer from github.com/robotgeek/Chip-E, then open the code in the Arduino IDE and upload it to the Geekduino board. Stack the Sensor Shield on top.

2. PREP THE SERVOS

The servos are your robot's hip and foot joints, so you need to set them to a specific position before assembling your robot or it won't work correctly. To center your servos at 90°, plug them into Sensor Shield's digital I/O pins 3, 5, 6, and 9, then run the Arduino sketch *centerServo.ino*. Now you can install the servo horns.

3. ASSEMBLE YOUR ROBOT

Bolt Chip-E together (Figure A) following the assembly guide at robotgeek.com.

4. CONNECT THE ELECTRONICS

Connect the servos, LCD, buzzer, and IR receiver to the Sensor Shield as follows, and you're done (Figure B).

Right Hip Servo	Digital 9
Left Hip Servo	Digital 10
Right Foot Servo	Digital 5
Left Foot Servo	Digital 6
LCD Screen	I2C
Buzzer	Digital 12
IR Receiver	Digital 2

5. PROGRAM AND CONTROL CHIP-E

Upload and run the *Chip-E_Gamepad* demonstration sketch and start moving Chip-E with the IR game pad (Figure C):

» Directional Pad: Walk forward or back, and turn left or right.

» TA and TB buttons: Speed up and slow down Chip-E's walking gait.

» A, B, Select, and Start buttons: Chip-E is very excitable; make him wiggle in place or dance!

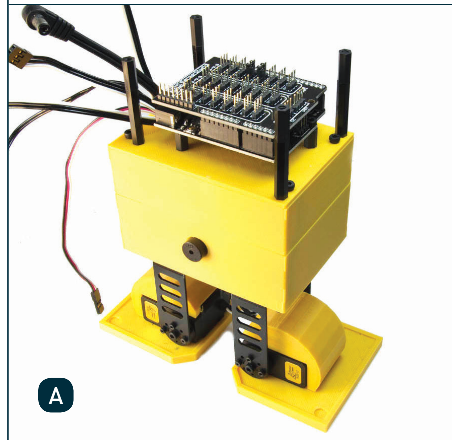
» A/B switch: Switch between 2 signal modes to run 2 robots, or to avoid crosstalk from another IR device.

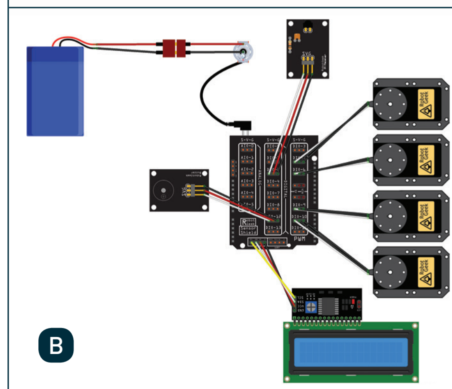
6. TUNE YOUR CHIP-E

If you notice Chip-E's legs are a bit off center and his feet aren't exactly parallel to the ground, you can fix this easily in code by adjusting a few values in lines 72–75 in the *Chip-E_Gamepad* sketch:

```
const int TRIM_RR = -5; //Trim on the right ankle
const int TRIM_RL = -7; //Trim on the left ankle
const int TRIM_YR = -4; //Trim on the right hip
const int TRIM_YL = -2; //Trim on the left hip
```

These values can be changed to any whole number, positive or negative, to adjust the center position of each servo. You can start by setting them all to zero, loading the code, and observing the default positions.


A


B

C

CUSTOMIZING YOUR BIPED

Chip-E's got room for more RobotGeek modules such as temperature and light sensors, IR transmitters, and LED lights. Customize the Arduino code to enable functionality unique to your liking. Print him in any colors you like, or download the Sketchup file and totally redesign him.

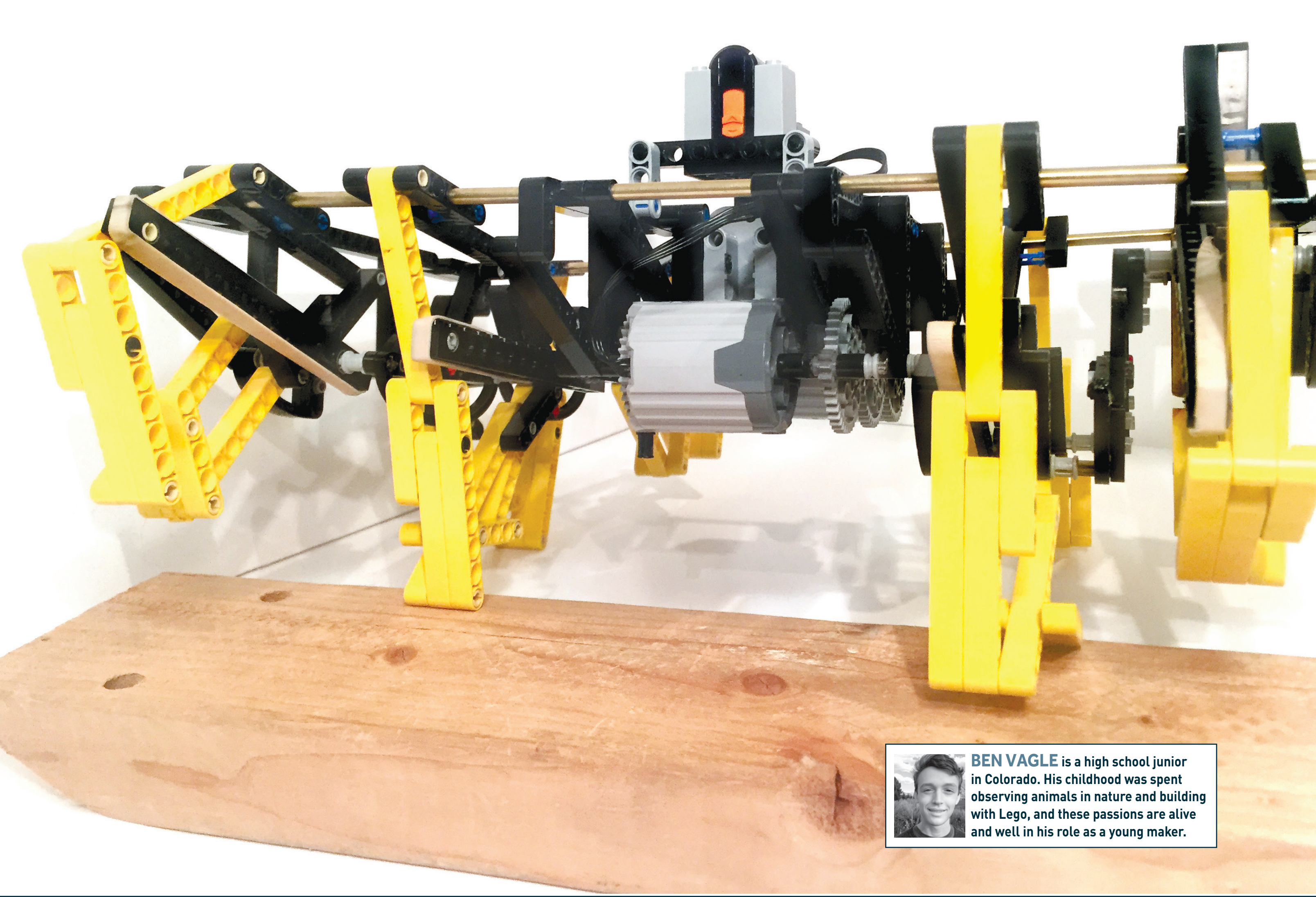
Woohoo! I hope everyone else likes these as much we do.

Watch Chip-E bust a move, and share your build at makezine.com/go/chip-e-biped-robot.

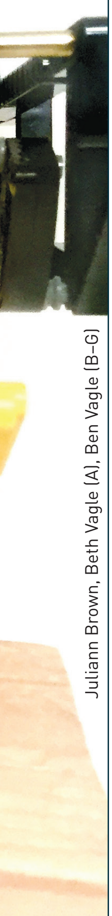
TrotBot
Better Than a Beest

Written by Ben Vagle

THIS 8-LEGGED WALKER BOT HAS A HIGH-STEPPING GAIT THAT DEFEATS OBSTACLES, AND YOU CAN BUILD IT WITH LEGO



BEN VAGLE is a high school junior in Colorado. His childhood was spent observing animals in nature and building with Lego, and these passions are alive and well in his role as a young maker.



THREE YEARS AGO, WHEN I WAS 13, I WAS PART OF THE TEAM THAT DEVELOPED A NEW TYPE OF WALKING ROBOT WE CALL "TROTBOT" using Lego Technic parts, and then helped to scale it up to the size of an SUV (Figure A). We showed off our latest giant TrotBot at World Maker Faire 2016 in New York, along with our Lego prototypes that are always a hit with the kids.

Out of this amazing experience, I've developed a walking robot STEM challenge that I'd like to share with other kids and educators.

TrotBot is a mechanical walker, like Theo Jansen's Strandbeest or Joe Klann's Mechanical Spider, but we developed it primarily to appeal to kids, to inspire them to learn some engineering and to want to build their own walking robots. Therefore we had two design goals:

1. Lifelike motion — TrotBot mimics a galloping horse
2. High functionality — we wanted kids to be able to really play with their robots and get lost in their imaginations as they walked them on the bumpy terrain found in nature, not just on flat man-made surfaces.

We reached our functionality goal by creating a mechanism that steps high, so its feet won't get stuck on bumps or obstacles. To show what I mean, here's my computer simulation (Figure B) comparing TrotBot's step (at right) to Strandbeest (left) and Klann's spider (center), showing how the different feet move when walking forward (right to left in the diagram).

But really you've got to see TrotBot in action, scrambling over a woodpile (Figure C) or slickrock in the desert (Figure D); check out my videos on YouTube (makezine.com/go/trotbot-youtube).

We learned a ton from scaling up TrotBot, and I've taken those insights back down to Lego scale to create the most functional walkers I've seen, with an improved foot design that increases ground contact while maintaining the high-stepping gait.

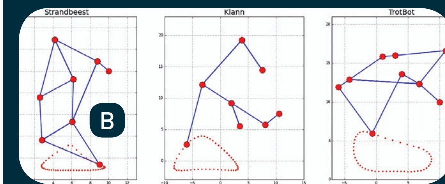
I could have made a robotics kit out of laser-cut wood, but I think I can make the biggest STEM impact by publishing the plans in Lego, a medium that most kids are already familiar with. Also, Lego's Mindstorms EV3 system has the basic automation components that walkers need, so kids and educators can use TrotBot as a platform to explore technology too. And once the designs are understood, it's easy to transfer them from Lego to other mediums, such as Vex components.

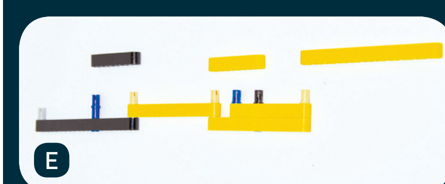
Until now, the specifications of TrotBot's mechanism haven't been made public, but I always get asked at Maker Faires if we would sell TrotBot kits, or at least show people how to make it. I've finally delivered on my promise to those people, and posted detailed plans as well as the engineering insights behind them.

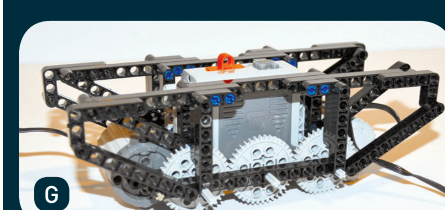
You'll find complete build instructions for this Lego TrotBot at my new website DIY Walkers (diywalkers.com/trotbot.html). It takes about one day of diligent work to make one. First you'll build each leg's mechanical linkage or "kinematic chain" (Figures E and F), then you'll mount the 8 legs in a frame with the motors and gear train to drive them (Figure G).

In addition to Technic beams, pins, and gears, this build uses Lego's Power Functions IR RX 8884 remote receiver, 8885 IR remote, 2 Lego Motors (I recommend the smaller 8883 M-Motors, but you can also use the half-speed, 4x torque XL 8882 motors), and an 8881 Battery Pack. I recommend lithium-ion AA batteries as they are lighter, last longer, and will improve walking performance. For the long axles connecting the legs to the frame, you can use Lego's plastic axles, but I prefer to use $^3/_{16}$" OD brass tubes, to better bear the robot's weight (aluminum rods are fine too). I'll document another version soon incorporating the EV3 Intelligent Brick.

My hope is that DIY Walkers inspires more kids and educators to play with engineering via designing and building walkers (and not just TrotBot; I'll be posting detailed plans on many walking mechanisms). Mechanical walkers can be a great incentive for students to master new skills, including design, kinematics, structural engineering, programming for control and optimization, project planning, and, trust me, *problem solving!*







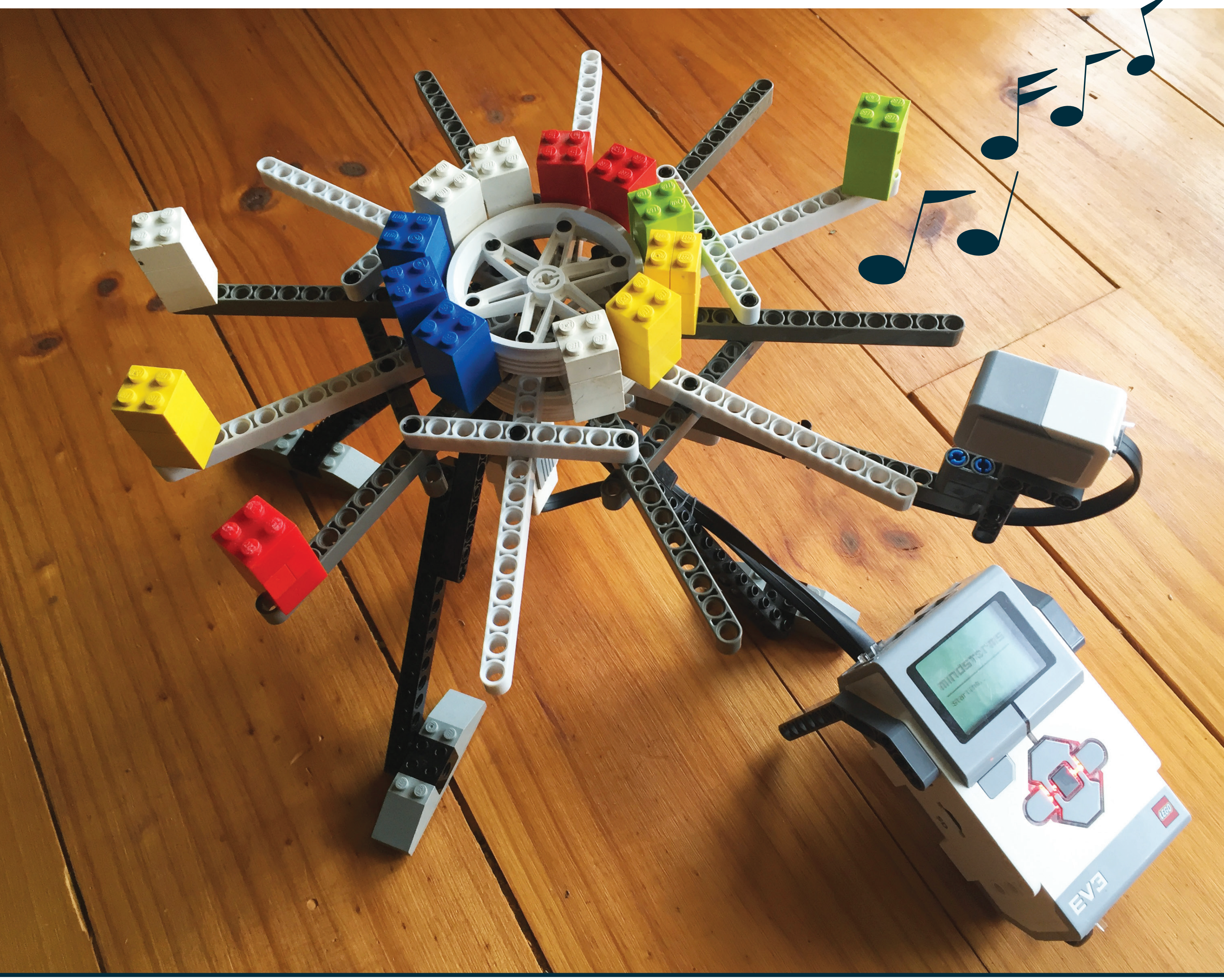
Juliann Brown, Beth Vagle (A), Ben Vagle (B–G)

Make your own TrotBot and share your builds, mods, and tips at makezine.com/go/lego-trotbot-8-legged-walker.

SPIN UP YOUR FAVORITE-COLOR LEGOS TO SEQUENCE SOUND SAMPLES IN ENDLESS COMBOS

Color-Sensing Sound Sequencer

Written by Brian McNamara



I'VE BEEN DESIGNING ORIGINAL ELECTRONIC INSTRUMENTS EVER SINCE I MADE A TOY SEQUENCER for my daughter's first birthday called the R-Tronic 8-Bit, and got it published in *Make:* in 2008. That one was a simple step sequencer that uses shapes to put sounds into a sequence that makes up a little song.

Since then I've built about 50 one-of-a-kind synthesizers, loopers, and sequencers, some that I've shared as DIY projects, and others that I sell on Etsy*, such as the Wicks Looper, the Automaphone, and the RotoSeq — a rotary sequencer that you program by placing (or removing) marbles in the path of 4 different light detectors.

This project is a Lego reproduction of my original RotoSeq that's simpler and smarter at the same time — it uses only a single sensor, but that sensor can detect 8 different colors. Little towers made of colored blocks are placed at the end of each arm, and as each block passes the color sensor, the sound that corresponds with that color is played. The variety of sounds and sequences is endless.

The Lego RotoSeq uses the Lego Mindstorms EV3 Intelligent Brick as a processor to do 3 things: control the motor that spins the arms, receive the data from the EV3 Color Sensor, then play the sounds from onboard WAV files. Of course you can easily substitute your own sound samples to create your own never-before-heard beats.

BUILD YOUR LEGO ROTOSEQ SEQUENCER

The RotoSeq is a really simple build. You'll construct a base with a motor to rotate a platform with 12 arms, and mount the color sensor where it can detect the position and color of the 2×2 brick towers. Then you'll hook up the EV3 brick for the brains.

1. BUILD THE BASE

Start by building the base around the Lego motor (Figure A). It doesn't matter how you do this, but it has to be sturdy, as the

Juliann Brown, Brian McNamara

TIME REQUIRED: 1–2 Hours
COST: $200–$350

MATERIALS

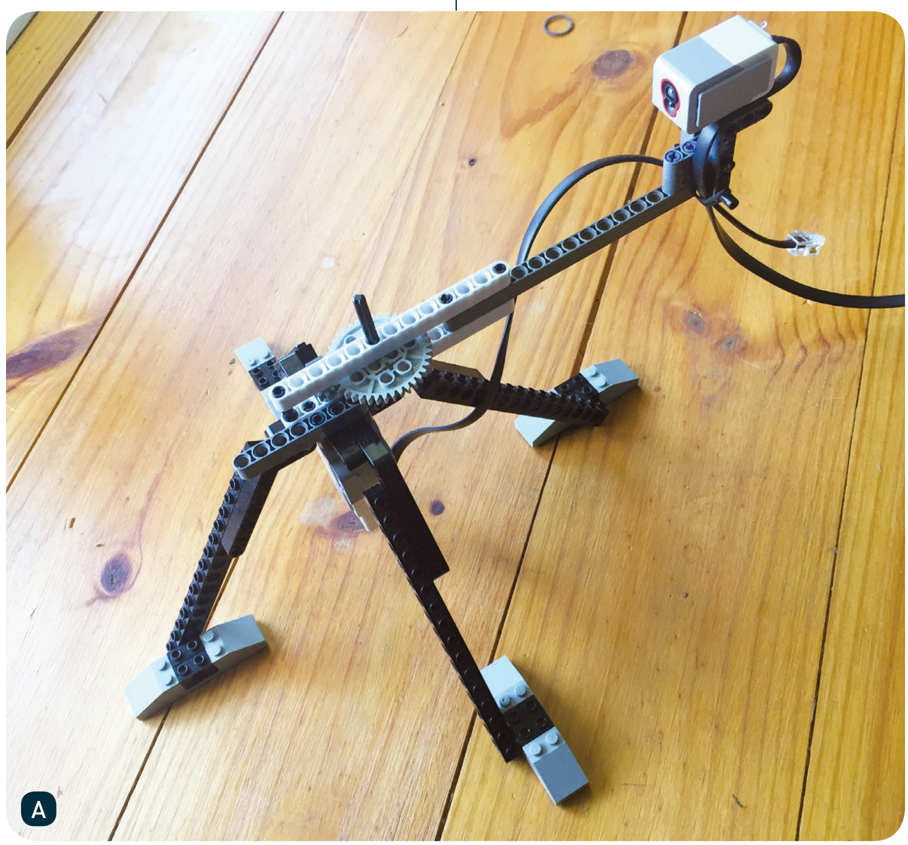
- **Lego Mindstorms EV3 Intelligent Brick**
- **Lego medium servomotor**
- **Lego EV3 color sensor**
- **Lego 2×2 bricks: blue, white, green, red, and yellow** Use 3 for each brick tower; you'll probably want at least 12 towers.
- **Lego gears: small (1) and large (1)**
- **Lego wheel (2)** for the hub — I used some from a Lego NXT set
- **Miscellaneous Lego bricks your choice,** to build the base and platform

TOOLS

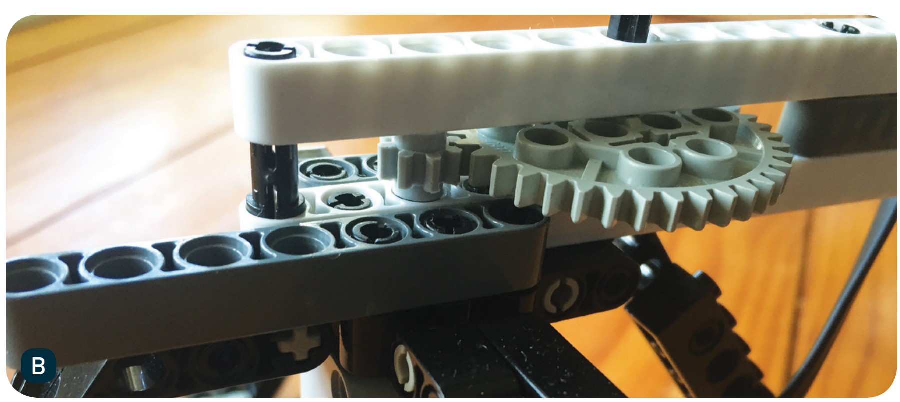
- **Computer with EV3 programming software** free download from lego.com/en-us/mindstorms

BRIAN McNAMARA builds a range of experimental instruments, but he is focused on autonomous sound-art devices and instruments that require unusual kinetic movements to be played. Many of these he also uses in his own musical performances as CupAndBow. etsy.com/shop/rarebeasts

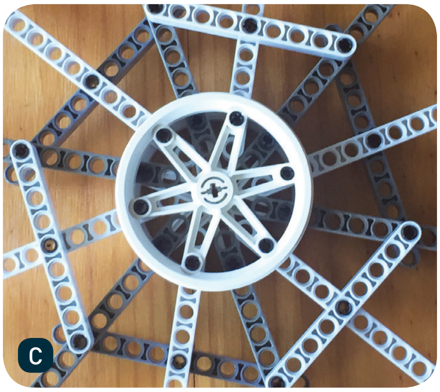
*Fun fact:
I've made a Beat Destructor for Adam Horovitz of the Beastie Boys and a Ruptutron for Thom Green from Alt-J.

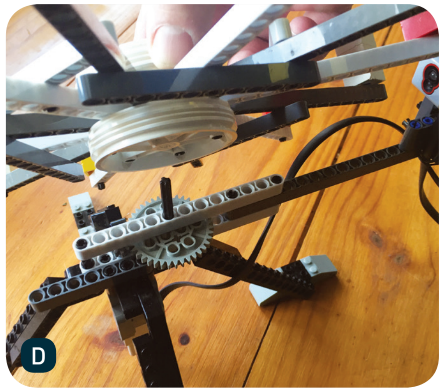


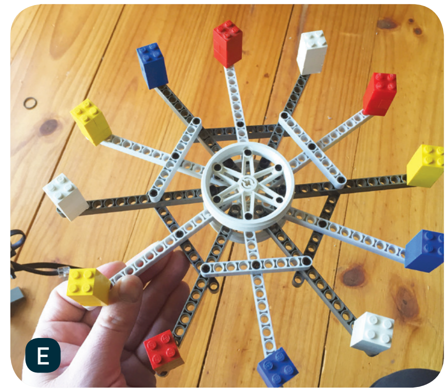
A



B





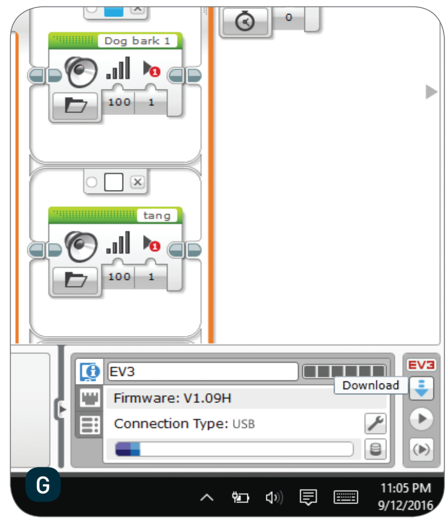


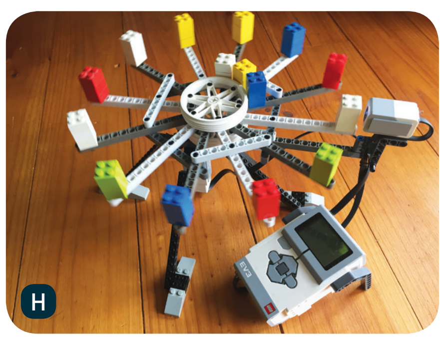
The First RotoSeq

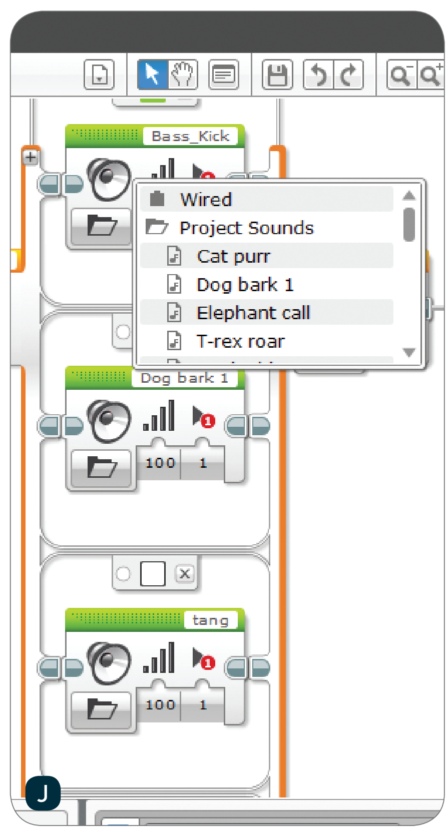
My original RotoSeq is powered by a hand-cranked generator. It's got 8 time slots and 4 sounds and uses 4 light sensors (light-dependent resistors or LDRs) connected to a Picaxe microcontroller to make sounds. In a way, it works oppositely to the Lego version — you start with a turntable filled with marbles in every position (no sounds) and then when a marble is removed from the turntable a sound is made that corresponds to that position. Each color represents one of 4 preprogrammed sounds, and each position represents one of 8 slots in the sequence.

At the heart of the RotoSeq is a Picaxe 28X processor, which takes information from the 4 LDRs, and generates the 4 lo-fi sounds in the sequence. LEDs provide a constant light source for the LDRs, and a modified servo is used to move the platform.

I still use the original RotoSeq, but the motor was a little noisy so I'm building a new version with some really nice woodwork and new samples. This one will probably be Arduino-based rather them Picaxe-based — and use a stepper motor.







Brian McNamara

platform rotates fast and can be heavy. I geared down the output with an 8-tooth gear on the motor and a 40-tooth gear on the platform side (Figure B, on page 45).

Also, the base needs to have a boom extending from one side to support the color sensor, long enough to reach just beyond the edge of the platform arms.

2. MAKE THE PLATFORM

The platform of the RotoSeq has 12 arms but it doesn't matter how you divide it up. For the hub, I used 2 wheels from a Lego NXT set which have 6 arms each; I put one on the top and one on the bottom and offset them to get 12 positions (Figure C).

Mount your platform's hub onto the motor's axle (Figure D).

3. BUILD YOUR COLOR TOWERS

Build several 3-high towers using 2×2 blocks in blue, red, white, green, and yellow. Fit a 2-unit long axle into the bottom of each one to allow it to sit in position at the end of the platform arms (Figure E).

4. MOUNT THE COLOR SENSOR

Now fit the color sensor into position so that it detects each 3-brick-high tower right in the middle (Figure F).

5. PROGRAM THE EV3 BRICK

Download the code file *Lego_Rotoseq.ev3* from the project page at makezine.com/go/lego-rotary-sequencer, then upload it to the EV3 Intelligent Brick using the Mindstorms EV3 software on your computer (Figure G).

Connect the color sensor to Input 1 and the motor to Output A (Figure H). You're done.

Special thanks to Edgar McNamara and Rhianna McNamara for all your help on this build.

NOW DROP SOME ROBOTRONIC BEATS

The RotoSeq is played by placing the colored Lego blocks onto the end of the rotating arms. Each color represents one of 5 sounds, and each position represents one of 12 slots in the sequence.

Now run the program on the EV3 brick. The up/down buttons control the speed of the spinning arms (Figure I), and the middle button stops the rotation to allow you to add and remove the colored brick towers.

Each color of brick is recognized by the software as the trigger for one of 5 specific sound samples. Just swap, delete, and remix the towers to rearrange your beats!

Tip: Store any unused towers on the platform, near the center where they won't be detected by the color sensor.

CUSTOMIZING THE SOUNDS

The sound samples can be easily modified using the Mindstorms EV3 software. To save your own samples, click Tools→Sound Editor, load your WAV file, click Save, then enter the name you want to call the sample.

Now find the programming block in the visual code that has a picture of a speaker on it. There will be 5 of these, labeled *Bass_kick*, *Dog_bark1*, *Tang*, *Coin*, and *Snare* (Figure J). Click on the name of the current sample you want to change, and add the one you want using the drop-down menu. Reload the program into the EV3 brick and you're ready to go.

Tip: I found my samples in an open - source WAV library by searching online for "open source drum samples."

GOING FURTHER

Want to use your RotoSeq as an input device for amps or for recording? The EV3 brick has no audio out jack. But with its built-in Bluetooth, Wi-Fi, and USB, there has got to be a way to get an output other than the speaker. I haven't found it yet, but maybe you can.

You should also be able to use more than 5 samples. Originally I was using all 8 colors that the sensor can recognize, but I was getting lots of false positive triggers, especially at higher speeds, so in the end I went with the most reliable color set. But I bet it can be done!

Have fun with this Lego project and please email me at rarebeasts.mail@gmail.com if you need any help or have any questions.

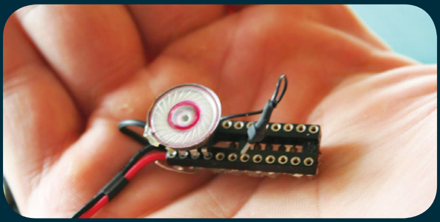
Watch the Lego RotoSeq in action, see more photos, and share your builds and mods at makezine.com/go/lego-rotary-sequencer.

More Great Synth Projects

by Brian McNamara at *Make:*

THE LUNA MOD LOOPER

Make a simple handheld synthesizer and looper box to generate intriguing sonic rhythms. makezine.com/projects/the-luna-mod-looper



THE MOOFTRONIC MINI SYNTH

Make new sounds from a truly tiny instrument based on a Picaxe microcontroller. makezine.com/projects/mooftronic-mini-synth

R-TRONIC TOY MUSIC SEQUENCER

Let babies play with shapes, sounds, and lights on this electronic music maker. makezine.com/projects/r-tronic-toy-music-sequencer

Written by Tim Deagan

HOW TO Hand Sew Leather

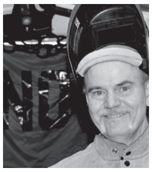
This basic technique will propel you toward custom tool coverings, bags, apparel, and more

Hep Svadja

THOUGH ITS HISTORY DATES BACK TO THE DAWN OF CIVILIZATION, leatherworking remains an enjoyable and useful skill even in the age of 3D printers. And while you can spend a lifetime learning the deeper intricacies of leather, the basics are easy enough for anyone to pick up. Among the most useful of these skills is the ability to sew pieces of leather together. The process is similar to sewing cloth, but has some significant differences. In this skill builder, we'll learn how to hand sew leather using the saddle stitch.

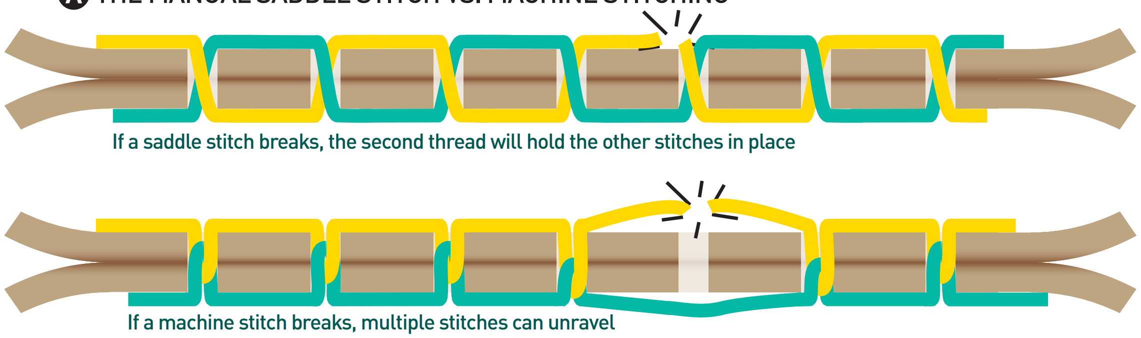
Hand sewing leather may seem daunting, but it's inexpensive, very strong, and less work than you might imagine. The saddle stitch is actually more durable than a machine stitch. When a machine stitch breaks, the entire piece will quickly unravel. When a saddle stitch breaks, the threads bind each other in place (Figure A).

Stitching needles are heavier, longer, duller, and have a larger eye than regular sewing needles. As opposed to sewing cloth, the needle is not intended to create its own hole. A hole is punched through the leather by an awl or chisel, then the needle is pushed through. We'll use two needles, one on each end of waxed thread. This thread is much heavier and stronger than cloth thread and is typically made from multiple cords of strong linen or synthetic material. A small lump of beeswax will help bind the thread (Figure B).



TIM DEAGAN (@TimDeagan) casts, prints, screens, welds, brazes, bends, screws, glues, nails, and dreams in his Austin, Texas shop. A career troubleshooter, he designs, writes, and debugs code to pay the bills. He's the author of *Make: Fire*, and has written for *Make:*, *Nuts & Volts*, Lotus Notes Advisor, and Database Advisor.

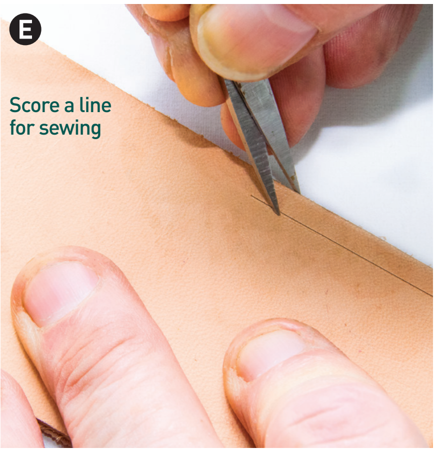
A THE MANUAL SADDLE STITCH VS. MACHINE STITCHING



B Basic hand sewing leather tools


C Piercing the thread with the needle


D Pull the thread to tighten the splice


E Score a line for sewing


F Punching the holes


G My homemade "stitching pony," a smaller version of a traditional stitching horse that holds leather while you sew

Cut some thread to the length of your arm span. For big projects, double that. Pass the end of the thread through the eye of the needle, then pierce the tip through the thread about 3" from the end (Figure C). Personally, I always pierce the thread twice.

Slide the thread down the needle until it passes the eye, then draw it tight (Figure D). Rub the beeswax along the splice and roll it tight between your fingers. Perform the same operations on the other end of the thread with the second needle.

Now let's prepare the leather. We need to score a line that is the same distance from the edge of the leather as the thickness of the two pieces of leather being sewn. There are fancy tools for doing this, but you can use scissors as a makeshift compass cutter to accomplish the same thing (Figure E).

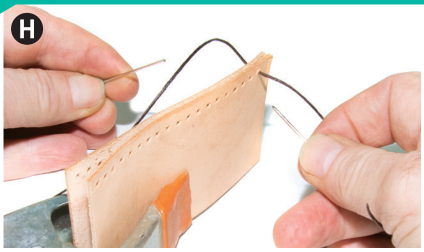
The distance between the holes varies with the intended use, thickness of thread, and weight of leather. If you're using an awl, an overstitch wheel is the best way to mark the locations. While using an awl is old school cool, chisel forks have become much more popular. Place the two pieces of leather together in the position you want to sew them. Set them on a smooth work surface padded with some thick scrap leather that you don't mind damaging.

Hold the fork along the marked line and use a soft-headed hammer to punch it through the leather. Drive the fork all the way through the two pieces being punched. Pull the fork out, set the first prong in the last hole and punch the next section (Figure F). Continue until holes are punched along the length to be sewn.

You can also hold the leather in a stitching horse, between your knees, in a soft-jawed vise, or in any manner that will leave your hands free (Figure G), because two needles require two hands.

Pass one of the needles through the hole where you want to start and pull it until

Tim Deagan

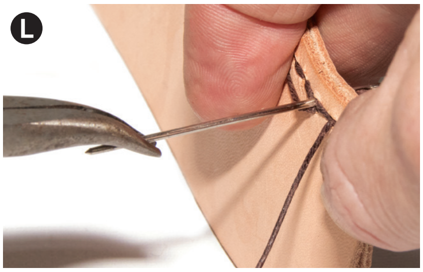


Ready to start sewing

Place the front needle in the hole

Pull the stitch tight

Backstitch to finish off



Pull a needle through tight stitches with pliers


Extreme close-up of the stitches

GOING FURTHER

What we've covered is only the most basic of stitches. If you'd like to learn more, the master reference is Al Stohlman's *The Art of Hand Sewing Leather*. This excellent instruction book has taught tens of thousands of leatherworkers basic and advanced techniques.

the workpiece is in the middle of the thread (Figure H).

Take the needle that will be on the back of your work and pass it back through the next hole. We will stitch toward ourselves. Pull two inches of thread through the hole. Take the front needle and push the tip just through the hole in front of the thread that's coming through. We always place the front needle in front of the thread from the back needle (Figure I).

Before you pull the front needle through the hole, we need to make sure it didn't pierce the back thread. If that happens, the stitch will have to be cut and you start over (or learn the advanced skill of dealing with a pierced thread). We can avoid this by pulling the back thread back through the hole as we push the front needle into the hole. When the front needle is almost all the way through we can stop pulling the back thread. Then we take a needle in each hand and pull evenly until the stitch tightens (Figure J). Note that the needles alternate front to back on each stitch.

Continue this sequence along the row of holes until you come to the end. To finish and secure the threads, we'll backstitch for two holes. This means that we will change direction and stitch over the last two stitches (Figure K).

The needles will be harder to get through the holes that already have thread in them. I generally end up using needlenose pliers to pull the needle through (Figure L). Be careful doing this since it may weaken or break the needle. You can avoid this by carefully pulling straight through and not putting any side force on the needle. Breaking off a needle can be dealt with if there's enough line to thread a new needle and keep going. Otherwise you'll have to backstitch as much as you can with the other needle and hope it holds.

Once finished, use small scissors or a utility knife to cut the remaining threads as close to the leather as possible. Many leatherworkers use a special tool called a stitching groover to gouge a shallow trough along the line of holes and, when finished sewing, use a hammer to tap the stitches down into the trough. This tucks them out of harm's way and makes them last longer.

With a little practice, hand sewing becomes a fast, easy, and fun way to make anything from a wallet to a saddle. Give it a try and discover a whole new world of leatherworking!

Tim Deagan

Making Sense of Multimeters

Discover the ins and outs of this essential electronics tool

Written by Charles Platt

Hep Svadja


A

B

C

D


E

Charles Platt

OF ALL ELECTRONICS TOOLS, I CONSIDER THE MULTIMETER TO BE THE MOST ESSENTIAL. It will tell you how much voltage exists between any two points in a circuit, or how much current is passing through the circuit. It will help you to find a wiring error, and can also evaluate a component to determine its electrical resistance — or its capacitance, which is the ability to store an electrical charge.

If you're starting with little or no knowledge, these terms may seem confusing, and you may feel that a multimeter looks complicated and difficult to use. This is not the case. It makes the learning process easier, because it reveals what you cannot see.

Before I discuss which meter to buy, I can tell you what not to buy. You don't want an old-school meter with a needle that moves across a scale, as shown in Figure A. That is an ***analog*** meter.

You want a ***digital*** meter that displays values numerically — and to give you an idea of the equipment available, I have selected four examples.

Figure B shows the cheapest digital meter that I could find, costing less than a paperback novel or a six-pack of soda. It cannot measure very high resistances or very low voltages, its accuracy is poor, and it does not measure capacitance at all. However, if your budget is very tight, it will work for basic projects.

The meter in Figure C offers more accuracy and more features. This meter, or one similar to it, is a good basic choice while you are learning electronics.

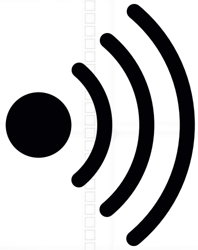
The example in Figure D is slightly more expensive but higher quality. This particular model has been discontinued, but you can find many like it, probably costing two to three times as much as the NT brand in Figure C. Extech is a well-established company trying to maintain its standards in the face of cut-price competitors.

Figure E shows my personal preferred meter at the time of writing. It is physically rugged, has all the features I could want, and measures a wide range of values with extremely good accuracy. However, it costs more than 20 times

CHARLES PLATT has written more than 50 books, the most recent being *Make: Tools*, which explains the use of hand tools to build more than 20 projects, with no prior experience necessary. His book *Make: Electronics* is a bestseller in its field, and he is the author of the *Encyclopedia of Electronic Components*. He is a contributing editor to *Make:* magazine.

F Ω Ω Ω

Three samples of the Greek symbol omega, used to represent electrical resistance.

G 

This symbol indicates the option to test a circuit for continuity, with audible feedback. It's a very useful feature.

as much as the lowest-priced, bargain-basement product. I regard it as a long-term investment.

How do you decide which meter to buy? Well, if you were learning to drive, you wouldn't necessarily need a high-priced car. Similarly, you don't need a high-priced meter while you are learning electronics. On the other hand, the absolute cheapest meter may have some drawbacks, such as an internal fuse that is not easily replaceable, or a rotary switch with contacts that wear out quickly. So here's a rule of thumb if you want something that I would regard as inexpensive but acceptable: Search eBay for the absolute cheapest model you can find, then double the price, and use that as your guideline.

Regardless of how much you spend, the following attributes and capabilities are important.

RANGING

A meter can measure so many values, it has to have a way to narrow the range. Some meters have ***manual ranging***, meaning that you turn a dial to choose a ballpark for the quantity that interests you. A range could be from 2 to 20 volts, for instance.

Other meters have ***autoranging***, which is more convenient, because you just connect the meter and wait for it to figure everything out. The key word, however, is "wait." Every time you make a measurement with an autoranging meter, you will wait a couple of seconds while it performs an internal evaluation. Personally I tend to be impatient, so I prefer manual meters.

Another problem with autoranging is that because you have not selected a range yourself, you must pay attention to little letters in the display where the meter is telling you which units it has decided to use. For example, the difference between a "K" and an "M" when measuring electrical resistance is a factor of 1,000. This leads me to my personal recommendation: I suggest using a manual-ranging meter for your initial adventures. You'll have fewer chances to make errors, and it should cost slightly less.

A vendor's description of a meter should say whether it uses manual ranging or autoranging, but if not, you can tell by looking at a photo of its selector dial. If you don't see any numbers around the dial, it's an autoranging meter. The meter in Figure D does autoranging. The others that I pictured do not.

VALUES

The dial will also reveal what types of measurements are possible. At the very least, you should expect:

Volts, amps, and ***ohms***, often abbreviated with the letter V, the letter A, and the ohm symbol, which is the Greek letter omega, shown in Figure F. You may not know what these attributes mean right now, but they are fundamental.

Your meter should also be capable of measuring milliamps (abbreviated ***mA***) and millivolts (abbreviated ***mV***). This may not be immediately clear from the dial on the meter, but will be listed in its specification.

DC/AC stands for direct current and alternating current. You may select these options with a DC/AC pushbutton, or choose them on the main selector dial. A pushbutton is probably more convenient.

Continuity testing is a useful feature that enables you to check for bad connections or breaks in an electrical circuit. Ideally it should create an audible alert, in which case it will be represented symbolically with a little dot that has semicircular lines radiating from it, as shown in Figure G.

For a small additional sum, you should be able to buy a meter that makes the following measurements, in order of importance:

Capacitance. The majority of electronic circuits require small components called capacitors. Because small ones usually don't have their values printed on them, the ability to measure their values can be important, especially if some of them get mixed up or (worse) fall on the floor. Very cheap meters usually cannot measure capacitance. When the feature exists, it is usually indicated with a letter F, meaning farads, which are the units of measurement. The abbreviation CAP may also be used.

Transistor testing is indicated by little holes labeled E, B, C, and E. You can plug the transistor into the holes to verify which way up the transistor should be placed in a circuit, or if you have burned it out.

Frequency is abbreviated as Hz.

• • •

TASTE THE POWER

Can you taste electricity? It feels as if you can.

WHAT YOU WILL NEED
- **9-volt battery**
- **Multimeter**

CAUTION No More Than Nine Volts!
This experiment should only use a 9-volt battery. ***Do not*** try it with a higher voltage, and ***do not*** use a bigger battery that can deliver more current. Also, if you have metal braces on your teeth, be careful not to touch them with the battery. Most importantly, never apply electric current from any size of battery through a break in your skin.

PROCEDURE

Moisten your tongue and touch the tip of it to the metal terminals of a 9-volt battery.

Do you feel that tingle? Now set aside the battery, stick out your tongue, and dry the tip of it very thoroughly with a tissue. Touch the battery to your tongue again, and you should feel less of a tingle.

What's happening here? You can use a meter to find out.

SETTING UP YOUR METER

Does your meter have a battery preinstalled? Select any function with the dial, and wait to see if the display shows a number. If nothing is visible, you may have to open the meter and put in a battery before you can use it — check the instructions that came with the meter.

Meters are supplied with a red lead and a black lead. Each lead has a plug on one end, and a steel probe on the other end. You

insert the plugs into the meter, then touch the probes on locations where you want to know what's going on. See Figure 1. The probes detect electricity; they don't emit it in significant quantities. When you are dealing with small currents and voltages, the probes cannot hurt you (unless you poke yourself with their sharp ends).

Most meters have three sockets, but some have four (see Figures 2 and 3). Here are the general rules:

One socket should be labeled COM. This is ***common*** to all your measurements. Plug the black lead into this socket, and leave it there.

Another socket should be identified with the ohm (omega) symbol, and the letter V for volts. It can measure either resistance or voltage. Plug the red lead into this socket.

The voltage/ohms socket may also be used for measuring small currents in mA (milliamps) ... or you may see a separate socket for this, which will require you to move the red lead sometimes.

An additional socket may be labeled 2A, 5A, 10A, 20A, or something similar, to indicate a maximum number of amps. This is used for measuring high currents.

FUNDAMENTALS: OHMS

You're going to evaluate the resistance of your tongue, in ohms. But what is an ohm? We measure distance in miles or kilometers, mass in pounds or kilograms, and temperature in Fahrenheit or Centigrade. We measure electrical resistance in ohms, which is an international unit named after Georg Simon Ohm, who was an electrical pioneer.

The Greek omega symbol indicates ohms, but for resistances above 999 ohms the uppercase letter K is used, which means ***kilohm***, equivalent to a thousand ohms. For example, a resistance of 1,500 ohms is equal to 1.5K.

Above 999,999 ohms, the uppercase letter M is used, meaning ***megohm***, which is a million ohms. In everyday speech, a megohm is often referred to as a "meg." If someone is using a "two-point-two meg resistor," its value is 2.2M.

A conversion table for ohms, kilohms, and megohms is shown in Figure 4.

A material that has very high resistance to electricity is called an ***insulator***. Most plastics, including the colored sheaths around wires, are insulators.

A material with very low resistance is a ***conductor***. Metals such as copper, aluminum, silver, and gold are excellent conductors.

MEASURING YOUR TONGUE

Inspect the dial on the front of your meter, and you'll find at least one position identified with the ohm symbol. On an autoranging meter, turn the dial to point to the ohm symbol as shown in Figure 5, touch the probes ***gently*** to your tongue, and wait for the meter to choose a range automatically. Watch for the letter K in the numeric display. Never stick the probes ***into*** your tongue!

On a manual meter, you must choose a range of values. For a tongue measurement, probably 200K (200,000 ohms) would be about right. Note that the numbers beside the dial are maximums, so 200K means "no more than 200,000 ohms" while 20K means "no more than 20,000 ohms." See the close-ups of the manual meters in Figure 6.

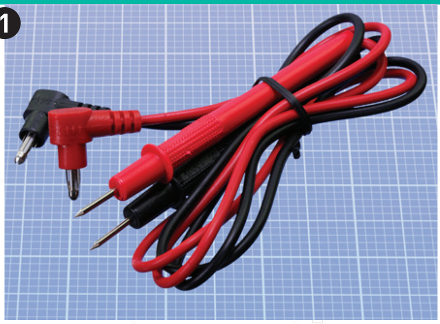
Touch the probes to your tongue about one inch apart. Note the meter reading, which should be around 50K. Put aside the probes, stick out your tongue, and use a tissue to dry it carefully and thoroughly, as you did before. Without allowing your tongue to become moist again, repeat the test, and the reading should be higher. Using a manual ranging meter, you may have to select a higher range to see a resistance value.

When your skin is moist (for instance, if you perspire), its electrical resistance decreases. This principle is used in lie detectors, because someone who knowingly tells a lie, under conditions of stress, may tend to perspire.

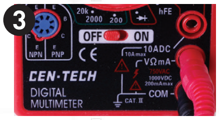
Here's the conclusion that your test may suggest. A lower resistance allows more electric current to flow, and in your initial test, more current created a bigger tingle.

CLEANUP AND RECYCLING

Your battery should not have been damaged or significantly discharged by this experiment. You can use it again. Remember to switch off your meter before putting it away. Many meters will beep to remind you to switch them off if you don't use them for a while, but some don't. A meter consumes a very small amount of electricity while it is switched on, even when you are not using it to measure anything.

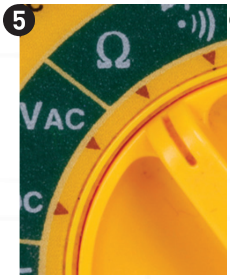
1


2


3


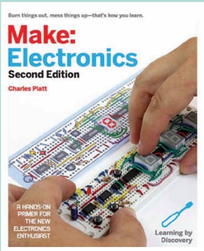
4

Ohms	Kilohms	Megohms
1Ω	0.001K	0.000001M
10Ω	0.01K	0.00001M
100Ω	0.1K	0.0001M
1,000Ω	1K	0.001M
10,000Ω	10K	0.01M
100,000Ω	100K	0.1M
1,000,000Ω	1,000K	1M

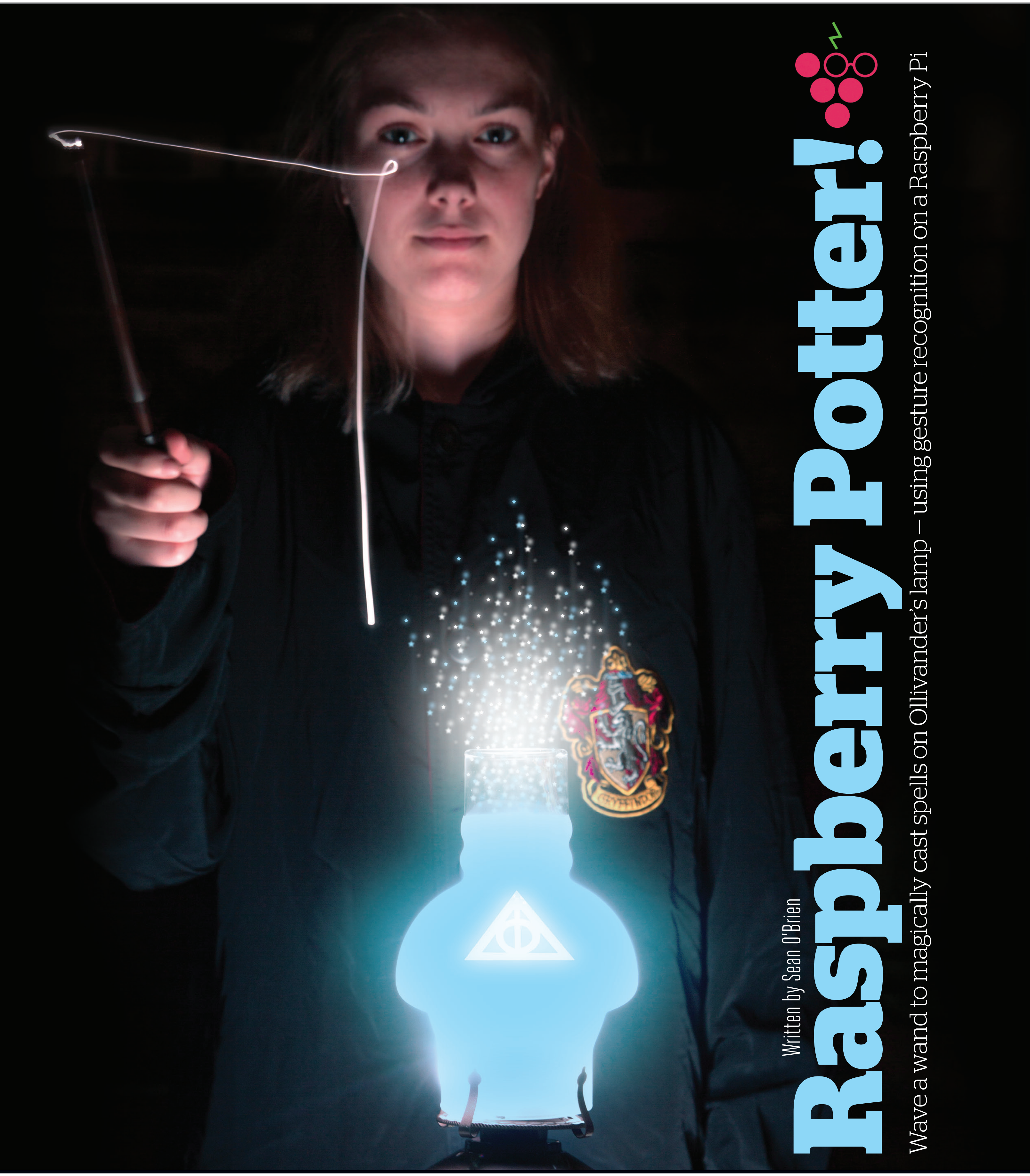
5


6


Charles Platt



This Skill Builder is excerpted from the 2nd edition of *Make: Electronics*, available at makershed.com and fine retailers everywhere. Visit makezine.com/go/multimeters to read more from the first chapter.



Raspberry Potter!

Written by Sean O'Brien

Wave a wand to magically cast spells on Ollivander's lamp — using gesture recognition on a Raspberry Pi

TO SAY HARRY POTTER IS A CULTURAL PHENOMENON IS AN UNDERSTATEMENT. It's simply part of our culture, arguably as beloved as any other media franchise in history, inspiring many of us to try to capture some of the magic on our own — which is what this project is about.

After a recent trip to Universal's Wizarding World of Harry Potter, my daughters and I decided to create a project that could use the interactive wands from the theme park, at home, to control our own props and gadgets. We called it Raspberry Potter because it was powered with a Raspberry Pi. We demonstrated the project last year at the Minneapolis/St. Paul Mini Maker Faire and this article — Ollivander's Lamp — is the latest extension of that project.

HERE'S HOW THE RASPBERRY POTTER WORKS:

1. Using an infrared camera, the Raspberry Pi computer looks for small circles of reflected infrared light in its field of vision.

2. These small circles of light get tracked for movement, using OpenCV computer vision software. These patterns of movement are the gestures or "spells," and you can make them using the reflective tip of a wand.

3. When the predefined patterns of movement are matched, a "spell" is cast (Figures A, B, and C) and the Raspberry Pi runs code that controls a connected device — in this case our magic lamp.

WHAT ABOUT THE WAND?

If you don't have an official interactive wand from the park (Figure D), don't worry! You can easily make your own wand by gluing a sequin to the end of a stick. Any wand-like object will work fine — it only needs a reflective tip that can be used to reflect infrared light. Just make sure the sequin is shiny and flat; the faceted ones don't work well. You can also use a "pearl sticker" such as Amazon #B001TNMW58.

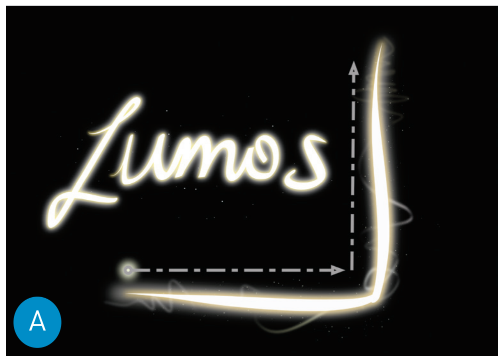
We 3D-printed a wood-grained wand from Thingiverse (thingiverse.com/thing:94309) and it turned out OK (Figure E). The Make Labs used some cool-looking ones from Etsy (etsy.com/listing/204449758), intended as party favors. Or check out raspberrypotter.com for a link to my new book with detailed instructions for making your own wand.

PREPARE THE ELECTRONICS

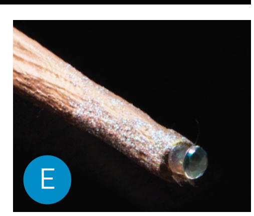
1. SET UP THE PARTICLE PHOTON

Let's start with the Particle Internet Button, which

Using wands to flush a toilet in the window of the Weasleys' Wizard Wheezes joke shop at Universal Studios Florida.



The tip of an interactive wand from Universal Studios.



The tip of a DIY magic wand that works just as well for this project.

Juliann Brown, Hep Svadja, Sean O'Brien

SEAN O'BRIEN is a marketing technologist with Minneapolis-based agency PadillaCRT.

Time Required: **A Week**

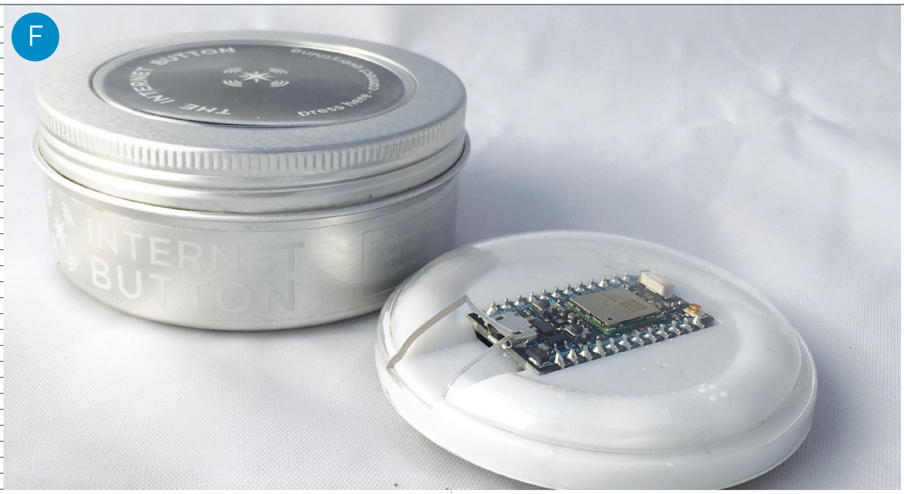
Cost: **$100-$200**

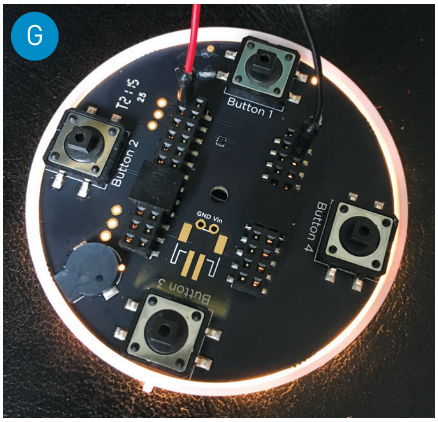
MATERIALS

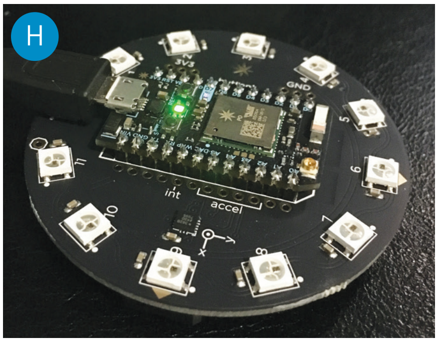
- **Hurricane lamp, clean and new** that can serve as an enclosure for a Raspberry Pi. I used Amazon #B001CSMHDM.
- **"Frosted glass" spray paint** for the lampshade
- **Raspberry Pi 3 Model B single-board computer**
- **MicroSD card, 8GB or larger**
- **Raspberry Pi NoIR Camera Module** I used the original but the V2 should actually work better, with better resolution.
- **IR filter film** to cover the camera lens. You can use old camera film negatives that are very dark, or buy IR-pass filter material such as Edmund Optics #43-952.
- **Particle Internet Button** includes a Particle Photon microcontroller, which can be easily removed and used for other projects, 11 RGB LEDs, and more.
- **Infrared (IR) LED** I used a 940nm LED, Adafruit #387, but other IR wavelengths will work. With a single LED, this project works best in the dark from 2 to 6 ft away — in brighter ambient light you'll need better filters and brighter IR too. For a demo during the day at Maker Faire I used an off-the-shelf IR security spotlight.
- **Micro-USB cable** to power the Pi
- **Jumper wires** female-female and male-male
- **Hookup wire (optional)** if you'd rather solder
- **Magic wand** preferably an official interactive wand from Universal Studios, but you can easily make your own using a sequin or "pearl sticker" (see "What About the Wand?").

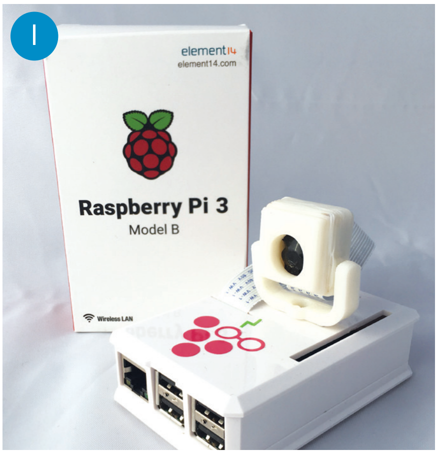
TOOLS

- **Hot glue gun**
- **Drill and twist bits** for drilling metal
- **High-speed rotary tool with cutting wheel** e.g. a Dremel
- **Locking pliers**
- **Hobby knife**
- **Soldering iron (optional)**









Pi and camera enclosures from the original Raspberry Potter project. You'll hide yours in Ollivander's Lamp instead!

will double as our light source (Figure F) — in addition to the Particle Photon microcontroller, it offers 11 bright RGB LEDs for all sorts of possible effects.

To be honest, using this device for our purposes is complete overkill — the Internet Button offers lots of other cool functionality like Wi-Fi connectivity, directional buttons, and a 3-axis accelerometer. But it's easy to control and fits perfectly with our hurricane lamp concept. Plus, it provides endless possibilities for extending this project in your own way. Let's set it up, add some code, and prepare to connect it directly with the Raspberry Pi.

1a. Follow the instructions to activate the Particle Internet Button at docs.particle.io/guide/tools-and-features/button/core.

1b. Once activated, grab the *lightsource.ino* code from my Github repo at github.com/sean-obrien/rpotter/tree/master/ollivanderslamp. Download the code and then flash it to the Internet Button through the online console at particle.io.

1c. We'll power the Internet Button directly through the Photon's breakout pins: Connect a wire to the 3.3V pin and another to GND as shown in Figure G. We'll also use the Photon's analog GPIO pins to control the lights; connect the A0, A1, and A2 pins to the Pi after setting up the Pi.

1d. Carefully remove the translucent plastic cover on the Internet Button. This will make the lamp just a little bit brighter. When done, it should look something like Figure H.

2. SET UP THE RASPBERRY PI

We'll need to add some software packages that give the Raspberry Pi (Figure I) the ability to utilize the camera, GPIO, and some basic computer vision. We'll assume you're running Raspbian Jessie with Pixel as your Pi's operating system. If you aren't, install it first, by downloading from raspberrypi.org/downloads/raspbian.

2a. Install OpenCV

Once the Pi is running and online, install OpenCV. This will probably take a few hours! OpenCV is an awesome open source computer vision project, and there's a great installation guide for the Raspberry Pi 3 at pyimagesearch.com/ 2016/04/18/install-guide-raspberry-pi-3-raspbian-jessie-opencv-3.

2b. Install PiCamera

Now make sure the camera module is ready to go. Physically install the Pi NoIR camera to the Pi by plugging in the ribbon cable. Then run the following command on the Pi:

```
sudo pip install --upgrade “picamera[array]”
sudo pip install imutils
```

2c. Install Pigpio

Finally, we need to install Pigpio to allow us to communicate via Python directly through the Pi's GPIO pins. Follow the instructions under "Method 1" at abyz.co.uk/rpi/pigpio/download.html. Then start up the Pigpio daemon by running:

```
pigpiod
```

2d. Install the Raspberry Potter script

Now we can download and install the Python script that runs our little project. Download the latest and greatest Raspberry Potter script from the *rpotter* repo on github.com/sean-obrien/rpotter.

Attach the Pi to a monitor or display and run the script to make sure it works properly. If everything is installed correctly you should see video from your camera on the screen.

2e. Set up the daemons

Now that we know it's running, we have to add a background process (an *init* daemon) so that the Raspberry Potter script will start automatically anytime we

Sean O'Brien

start up the device.

Go to github.com/sean-obrien/rpotter/tree/master/ollivanderslamp and download *rpotter-startup*. Install this file to the */etc/init.d/* directory and make it executable by running:

```
sudo chmod 755 /etc/init.d/rpotter-startup
```

Then register the script to run at startup:

```
sudo update-rc.d rpotter-startup defaults
```

Now unplug the Pi from the display and restart it. It should start up the script automatically and start looking for something that looks like a magic wand.

3. CONNECT THE PI TO THE PARTICLE

Now that our Raspberry Pi is ready to go, we'll wire the Pi to the Photon. Figure J is a wiring diagram showing how to connect the two. You'll also notice the commented pin references in the Photon code we downloaded earlier. Basically, connect pins A0, A1, and A2 on the Photon to pins 15, 16, and 18 respectively on the Pi. Make sure that 3.3V and GND from the Pi are connected to the Photon as well — otherwise it won't get power!

When we're done, it should look something like Figure K. Figure L shows the Raspberry Pi successfully powering the Internet Button!

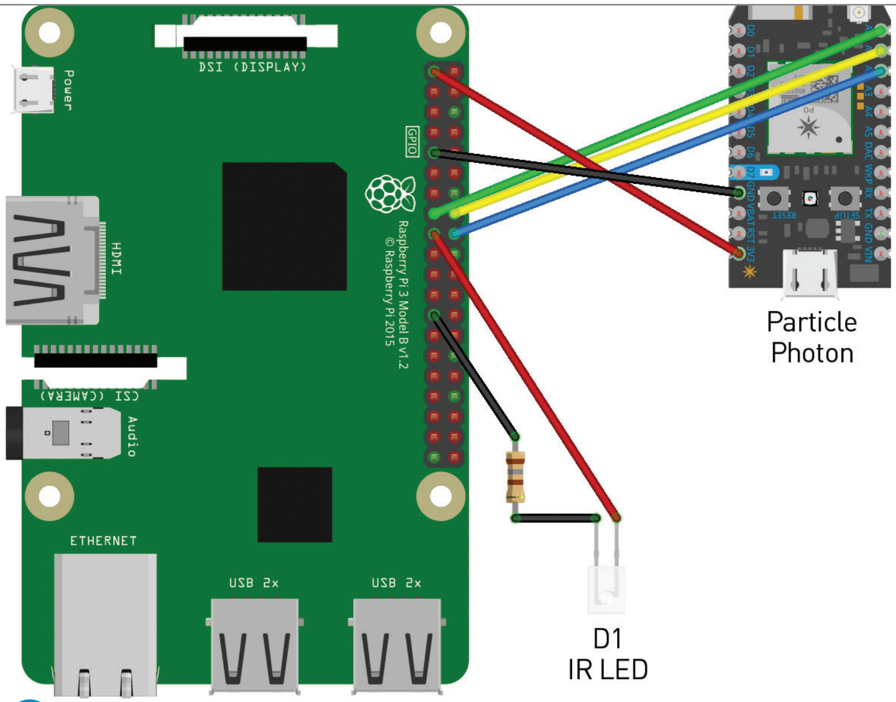
4. PREPARE THE IR LED

Figure J also shows an infrared LED, which we'll connect to the Pi later. Using an IR LED for illumination allows the Raspberry Potter to work in low-light conditions and even in the dark. Follow Figure J to add a resistor and power leads to the IR LED. I lazily used jumper wires (Figure M); you may want to solder yours.

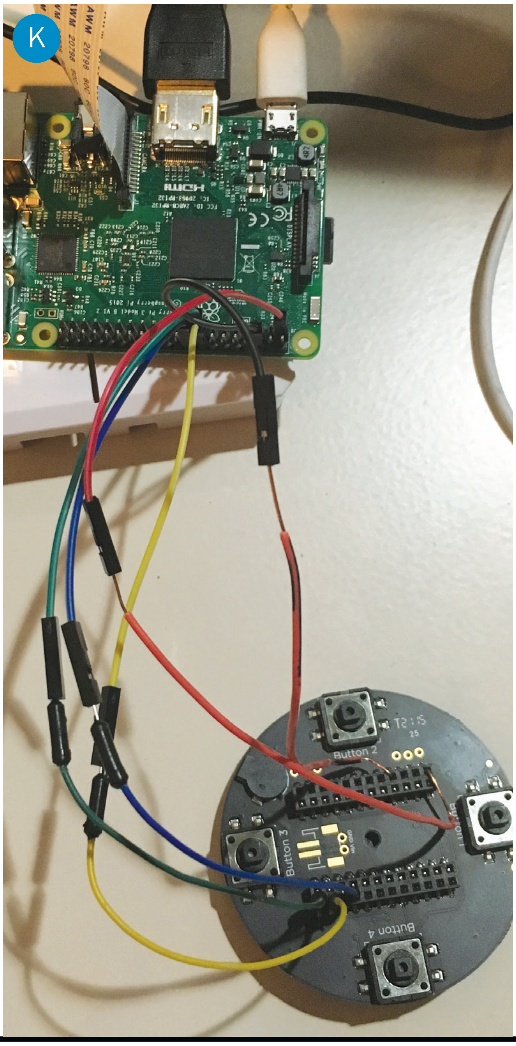
You'll need to choose a resistance value based on your particular LED. Check out ledcalculator.net to determine what resistor you'll need.

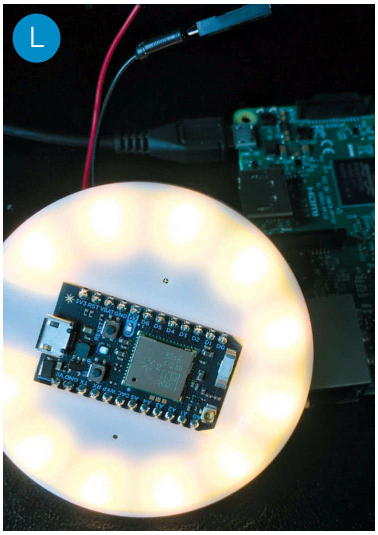
CREATE THE LAMP

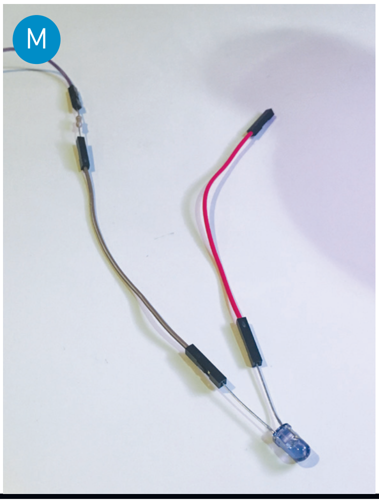
In the movies, Ollivander's lamp looks a bit like an antique hurricane lamp, so that's what I've based this build on. Feel free to use any kind of lamp, of course — the one I used can be found on Amazon, item #B001CSMHDM (Figure N). We also love

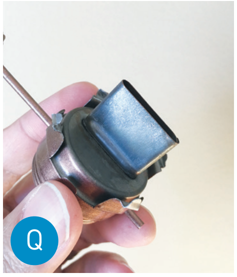


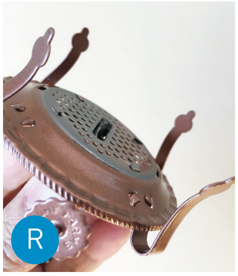
J Raspberry Pi 3 (RPI-3-V1.2)

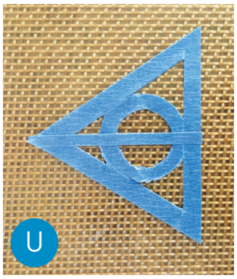

K


L


M







the Victorian hurricane lamp model by Louise Driggers (thingiverse.com/loubie) but we haven't tried to fit this project into it!

5. MODIFY THE WICK CHANNEL

It just so happens that our Particle Internet Button perfectly fits inside the lamp glass, and the base is large enough to fit our Pi — but how to wire the two together?

If we unscrew the burner, we can remove the wick (Figure O), which gives us a hole to thread wires through.

So far so good, but we also want to remove the wick channel (Figure P) so that the Internet Button can sit with a lower profile on the burner.

Remove the top of the burner, and then cut off the top of the wick channel (Figure Q) with a rotary tool, or bend it back and forth with locking pliers until it snaps off. Then reattach the collar to the burner (Figure R).

Now we'll be able to thread the wires from the Internet Button through the collar into the base below.

6. FROST THE LAMP GLASS

Giving the lamp a frosted look will diffuse the Internet Button's full-color LED light and make the lamp really glow. I just used standard Rust-Oleum frosted glass spray (Figure S) to get the effect I wanted.

Tape off the top of the lamp glass to protect it from chips or scratches while handling it (Figure T).

The spray takes about 5–10 minutes to dry per coat. After 5 coats, I made a stencil of the Deathly Hallows logo just so we know what universe we're in (Figure U).

Of course, this would be a great time to apply some creativity — maybe do a stencil of your Patronus, or a lightning bolt. Apply your stencil to the frosted glass, and coat another 5 times. Here's what it looks like after all 10 coats (Figure V). Looking good!

7. MODIFY THE BASE

Drill 2 holes next to each other into the base of the lamp (Figure W): a larger hole for the camera lens, and a smaller hole for the IR LED. Make the holes close together, but leave enough room so that the camera board doesn't block the hole for the LED. These holes in the picture are almost too close together.

Now you need to open up the base of the lamp so you can insert the electronics. Cut through the bottom with a rotary tool (Figure X). We're lucky — it's just barely big enough for our Pi.

For ease of access and installation, you may want to go further and cut a bit more out of the base. Then crimp the edges using pliers, so that there aren't any sharp edges exposed (Figure Y).

8. MOUNT THE IR LED AND CAMERA

Take the IR LED that we wired earlier and thread it through the smaller hole in the base (Figure Z). Secure the LED in place with hot glue, but first test-fit the camera board to make sure you leave enough space for it (Figure AA).

We need to block as much visible light from the camera as possible to allow for more accurate infrared detection. Cut a small piece of developed film or IR-pass filter plastic and cover the lens of the camera board (Figure BB).

Secure the camera inside the base of the lamp with its lens looking out the hole, using hot glue (Figure CC). Be careful not to burn the board or any of the wiring!

9. PUT IT ALL TOGETHER

Thread the Internet Button through the collar and then attach the collar to the base (Figure DD).

Attach the wires from the Internet Button to the Pi as shown in the wiring diagram. Attach the LED and the camera board to the Pi as well (Figure EE). Fit all the wires and the Pi into the base and connect power to the Pi. And you're done!

You now have your very own magic lamp.

NOW FOR SOME WITCHCRAFT AND WIZARDRY

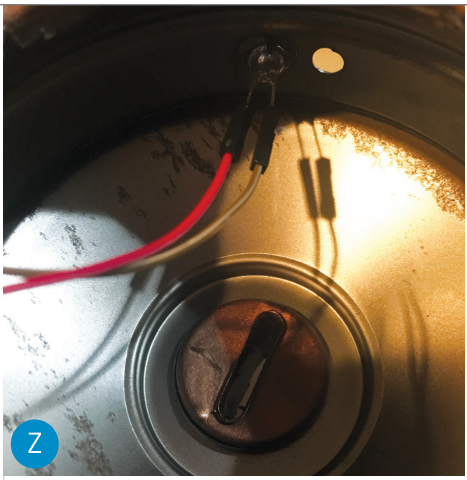
Inside Ollivander's lamp, the OpenCV software is watching and waiting to track the shiny end of your magic wand. Point your wand at the lamp, then move the wand right and then up to cast the "Lumos" spell and turn the lamp on (Figure FF).

Move your wand right and then down to cast "Nox" and turn the lamp off (Figure GG).

Move your wand left and then up to cast "Incendio" for a fire effect (Figure HH).

Practice your spells in the mirror!

Sean O'Brien, Hep Svadja



MAKING THIS PROJECT YOUR OWN

Now that you've done all the hard work, there's so much more you can do to make this project your own:

» **Create new light effects** by modifying the *lightsource.ino* code. To start experimenting with colors, just replace the RGB values in the `b.ledOn` function with any number from 0 to 255, like this:

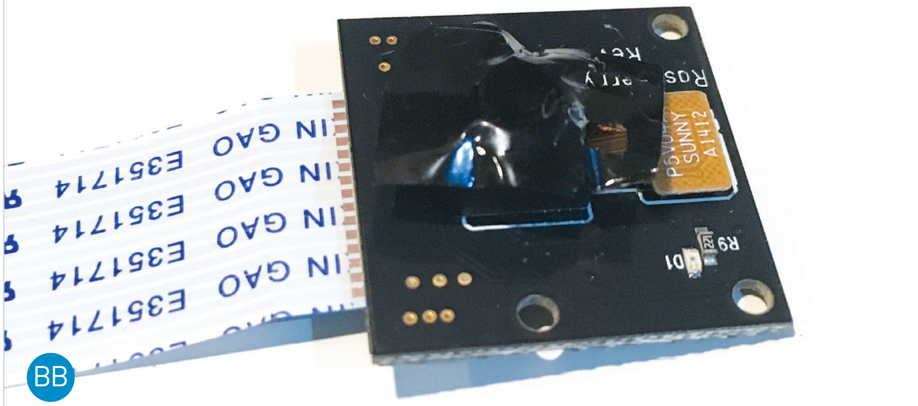
```
b.ledOn(i, 0,255,255);
```

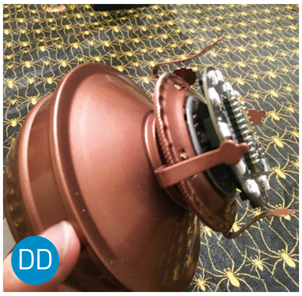
This line should make all the LEDs a light-blue color. Keep experimenting to get the effects you want: strobe lights, rainbows, holiday themes, it's all possible!

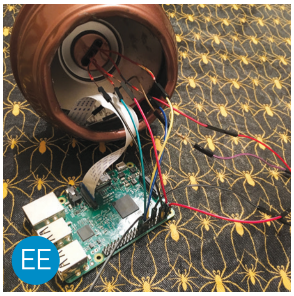
» **Add spells** Open up the *rpotter.py* script and add a gesture to `IsGesture()`, add a spell to `Spell()`, and add the corresponding pin on your Pi (copy lines 55–56 and modify for your new pin). If you've created something cool, send me a picture and let me know!

» **Connect more props** If you're feeling ambitious, and want to try more advanced effects — say you want to switch on a fan, or trigger a servo to open a box — you can hook your lamp up to low-voltage devices through the Pi's GPIO pins, or higher-voltage devices through a relay like a PowerSwitch Tail II.

Whether you're a wizard or a muggle, I hope you've enjoyed this project — it's been so much fun to create. As a bonus, check out raspberrypotter.com. You can contribute your own code, and I'd love to see your ideas for new effects or improvements. You'll also find links to other code examples and Raspberry Potter projects to try.







See more photos and share your Raspberry Potter builds at makezine.com/go/raspberry-potter-ollivanders-lamp.

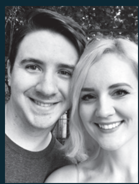
Written by Nick and Shayna Brewer

GIF It To Me

Build the PIX-E, the world's funnest dedicated GIF camera!

Hep Svadja, Nick Brewer

makezine.com/55



NICK AND SHAYNA BREWER founded MoonTower Labs (moontowerlabs.com) in Austin, Texas, in 2016. Nick is a hacker/maker who enjoys transforming antiques with modern electronics; Shayna is a digital artist who takes a beautiful, subtle, and humorous approach to her work.

Time Required: **5-6 Hours**
Cost: **$75-$90**

MATERIALS

- **Raspberry Pi Zero single-board computer, Version 1.3** the one with the camera connector
- **MicroSD card**
- **Raspberry Pi Camera module with cable**
- **3D printed case parts** Download the files for printing from thingiverse.com/thing:1761082.
- **LiPo battery, 3.7V, 2200mAh or 2500mAh** Adafruit Industries #1781 or 328, adafruit.com
- **Adafruit PowerBoost 500C charger and power supply** Adafruit #1944
- **Pushbutton switch, 16mm, illuminated** Adafruit #1439
- **USB Wi-Fi card, very small (optional)** such as Amazon #B003MTTJOY
- **USB drive, very small (optional)** such as Amazon #B00LLEN5FQ
- **On-off switch** I used a cool rocker switch, SparkFun #COM-08837. You can also use a small slide switch (we designed case options for both).
- **USB OTG adapter (optional)** to connect Wi-Fi card or USB drive to Pi Zero
- **Machine screws: M2 × 6mm (4) and M2.5 × 10mm (10)**
- **Hookup wire**
- **LED, 5mm** such as Adafruit #297
- **Resistors, 56Ω (2)**
- **Cellphone lens kit (optional)** such as Amazon #B00XVECB6S

TOOLS

- **Soldering iron and solder**
- **3D printer (optional)** You can print the files yourself or send them out to a service; see makezine.com/where-to-get-digital-fabrication-tool-access to find a machine or service you can use.
- **Multimeter** to test connections
- **Wire strippers**
- **Screwdriver**
- **X-Acto knife**
- **Third hands tool, Panavise, etc.**
- **Hot glue gun**
- **Heat gun**

I WAS ASKED BY MY FRIEND MATT GRIFFIN TO MAKE A 3D-PRINTED PROJECT FOR THE ULTIMAKER BOOTH at World Maker Faire 2016, and together we landed on a GIF camera. My thought process was that I wanted to create a camera that recalled those days when disposable cameras were a thing.

So my wife Shayna and I tackled the project: I did the soldering and printing, and she helped with the design and created the decorative wrappers that go over the camera. You can print and cut them out to give your PIX-E an extra bit of flair.

The result? A fully customizable 3D-printed camera that takes short GIFs using a Raspberry Pi Zero and Raspberry Pi Camera. *Hackaday* called it "the greatest technical achievement of our time, for the creation of content in the greatest artistic medium." We think they were joking, but we still love it.

Everything about this camera is designed to be modified:

- Want the body of the camera to be a different shape? Play around with the 123D Design files.
- Want longer GIFs or a different exposure? Dig into the code!
- Should it upload to Twitter or straight to an SD card or USB drive?
- A 2200mAh LiPo battery will give you around 7 hours of use; 2500mAh will give you around 8 hours.

Anyone with intermediate making/printing/coding skills should be able to put this together pretty easily. Here's how to make it.

1. PREPARE THE 3D-PRINTED CAMERA BODY

3D print the camera parts from thingiverse.com/thing:1761082, or have them printed, in the colors of your choice.

Then use an M2.5 screw to gently tap threads in the posts on the 3D-printed parts. Don't tighten these down too tight or you'll strip the hole. This step will make it much easier to assemble the camera later.

2. INSTALL THE SOFTWARE

It's recommended that you set up the SD card on the Raspberry Pi Zero before you put it into the camera. Then update the software packages on your Pi Zero by running:

```
sudo apt-get update
sudo apt-get upgrade
```

Next, install PiCamera (picamera.readthedocs.org), GraphicsMagick (graphicsmagick.org), and Gitcore:

```
sudo apt-get install python-picamera
sudo apt-get install graphicsmagick
sudo apt-get install git-core
```

Now install the PIX-E project code that I wrote, called GifCam:

```
sudo git clone github.com/nickbrewer/gifcam.git
```

If you want your PIX-E camera to upload GIFs directly to Twitter, install Twython (github.com/ryanmcgrath/twython).

Or, optionally, mount a USB flash drive (raspberrypi-spy.co.uk/2014/05/how-to-mount) if you want to save the GIFs there (instead of the microSD card). Use the USB OTG adapter to connect it.

3. CREATE THE AUTORUN SCRIPT

Run the command:

```
sudo crontab -e
```

and then add this line to the end of that file:

```
@reboot sh /home/pi/gifcam/launcher.sh
```

The launcher.sh script is set up to run the basic *gifcam.py* code, which saves GIFs to the Pi Zero's microSD card. If you want to use Twitter or a USB drive instead, you'll have to modify it as follows:

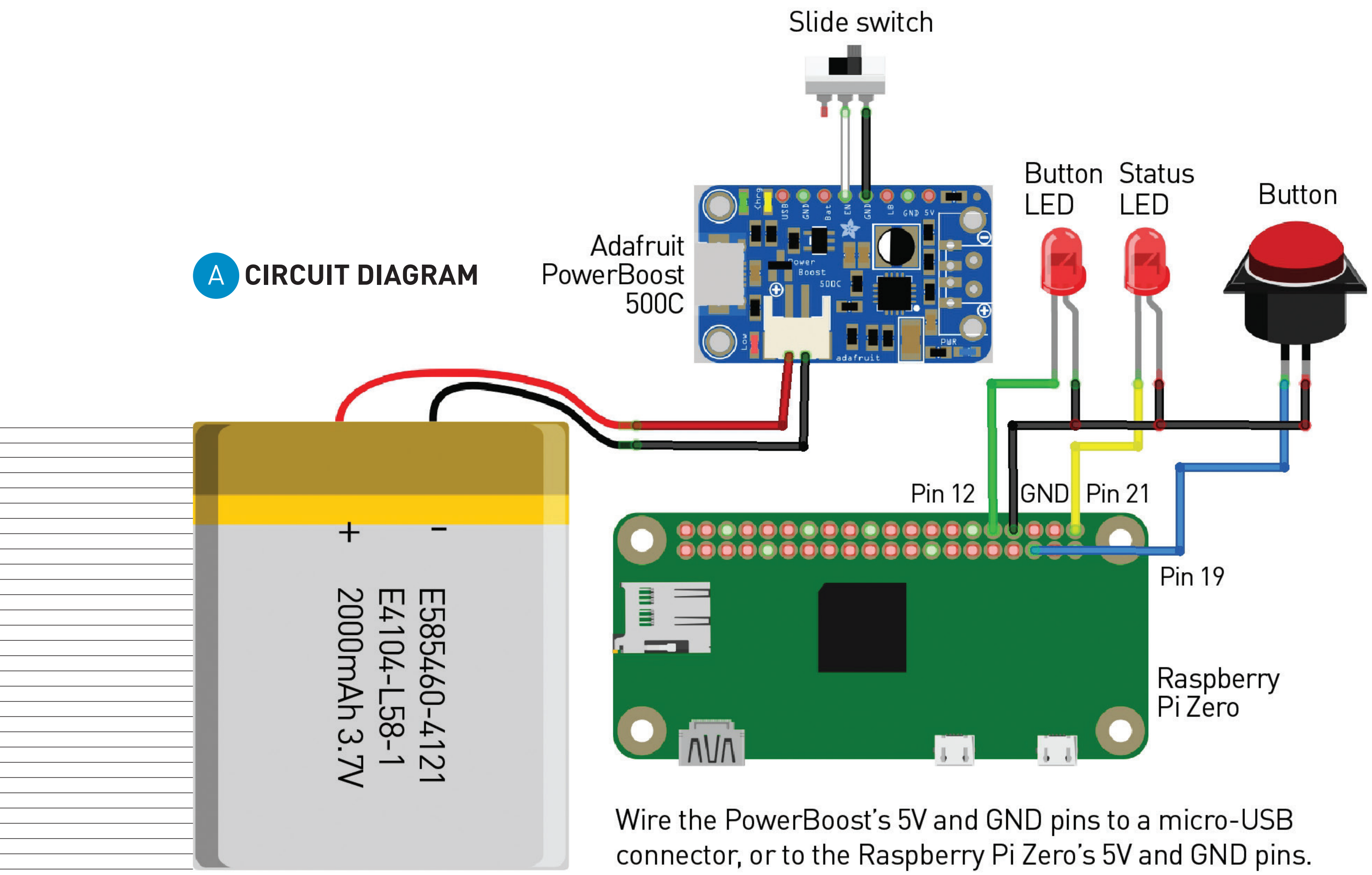
- To save GIFs to Twitter, edit launcher.sh and change line 7 to:
  `sudo python gifcamtwitter.py`
- To save GIFs to the USB drive, edit launcher.sh line 7 to:
  `sudo python gifcamusb.py`

If you're hitting "permission denied," run the command:

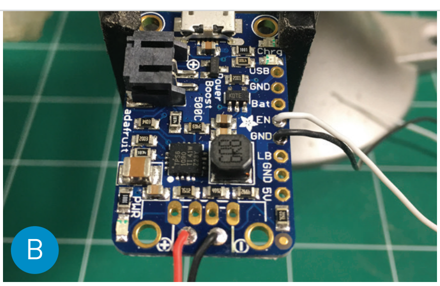
```
sudo chown -R pi /home/pi/gifcam/
```

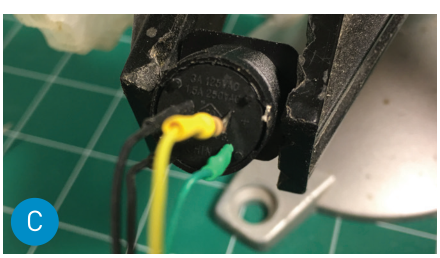
That's it. The software setup is complete.

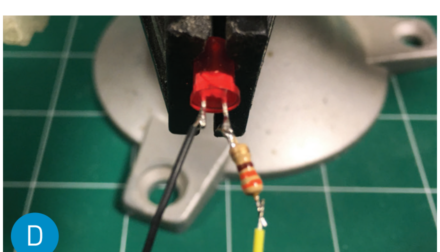
A CIRCUIT DIAGRAM



Wire the PowerBoost's 5V and GND pins to a micro-USB connector, or to the Raspberry Pi Zero's 5V and GND pins.


B


C


D

4. WIRE THE ELECTRONICS

You'll wire the electronic components separately, then assemble them into the camera body, and finally solder them to the Raspberry Pi Zero after it's installed.

Following the wiring diagram (Figure A), first wire the PowerBoost 500C as shown (Figure B).

Next wire the illuminated pushbutton (don't forget to add a 56Ω resistor to its LED) (Figure C).

Then wire the separate status LED that will attach to the camera back. Add its resistor to the anode (longer) leg (Figure D).

5. ASSEMBLE YOUR GIF CAMERA

Attach the Raspberry Pi Camera Module to the front of the camera (Figure E), then attach the front to the camera body using the M2 screws. Make sure the ribbon cable is oriented correctly and use caution; it's fragile.

Attach the 2500mAh LiPo battery to the camera body using hot glue (Figure F). If you use a 2200mAh cylinder LiPo instead, it will snap into the right side of the body.

Then attach the PowerBoost 500C power supply board to the case using M2.5 screws (Figure G). *Do not* plug the battery into the power board just yet; save that for the end.

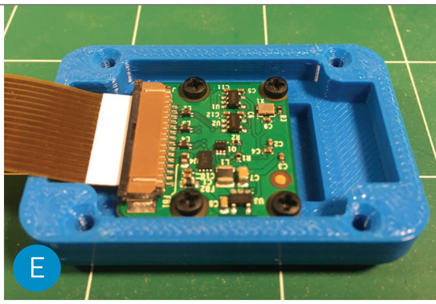
Attach the ribbon cable to the Raspberry Pi Zero and mount the Zero in the case using M2.5 screws (Figure H). Install the pushbutton on top as well.

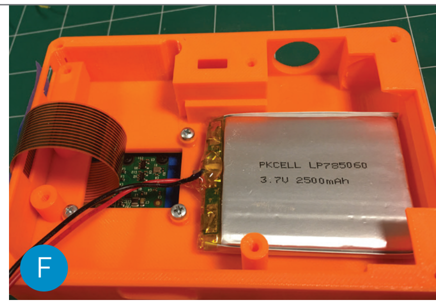
Wire the pushbutton, power board, and status LED to the Pi Zero as shown in the wiring diagram. (You can wire the PowerBoost's 5V output (the pins marked + and –) to a micro-USB connector to power the Pi, or wire it directly to the Pi's GPIO pins (+5V and GND) pins; I used the micro-USB connector.)

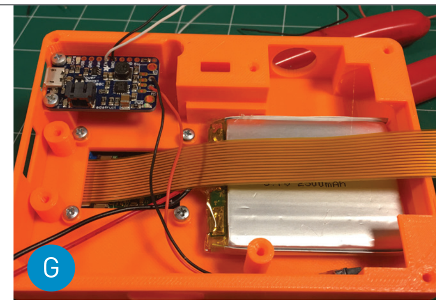
Wire the PowerBoost's GND and EN pins to the slide switch on the back of the camera, and hot-glue the status LED into place (Figure I). Don't go overboard here, a little will do.

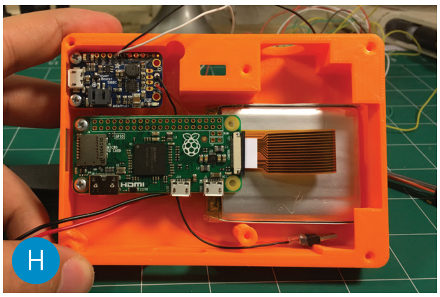
Make sure the switch is off and plug the

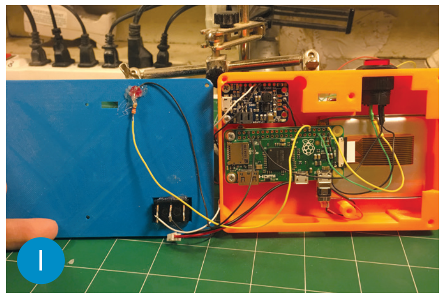
Nick and Shayna Brewer

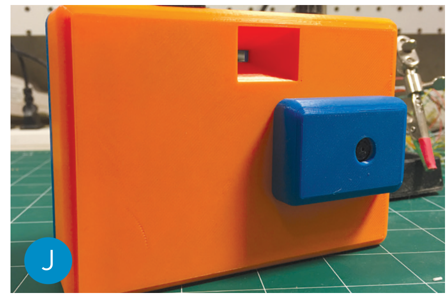


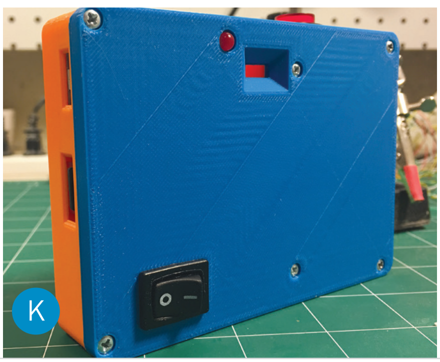


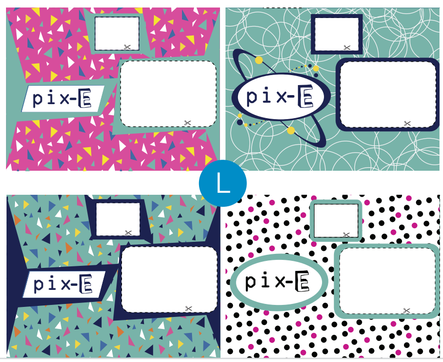


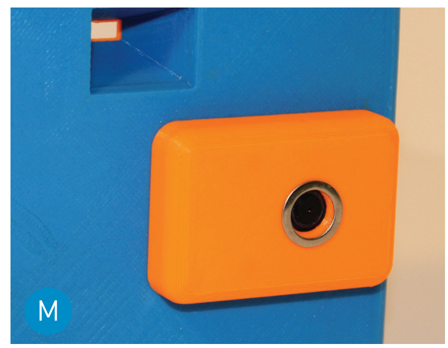












battery into the PowerBoost board. Nothing should happen. If you turn the switch on, the board's blue ON light should come on.

If you'd like to automatically save your GIFs to Twitter, plug the USB Wi-Fi card into the Pi Zero, using the USB OTG adapter. If you're saving GIFs to the USB thumb drive make sure the drive is plugged into the Pi Zero, using the USB OTG adapter.

Test it out and if everything's working, close it up using M2.5 screws (Figures J and K), and go get some GIFs.

NOW GIF TIL IT HURTS

Your PIX-E GIF camera charges up quickly through the Adafruit PowerBoost's micro-USB port on the left side of the camera. On a full charge it should last about 7 hours of continuous usage.

Here's how to use it:

- » Turn the PIX-E on
- » Wait for the red status LED to begin blinking on the back of camera
- » Aim the camera at your subject matter and hold down the top button to begin filming
- » The blinking LED will turn solid for 4 seconds (recording) and then shut off (processing)
- » Wait for the LED to begin blinking again, and repeat.

The GIFs will be saved on the Pi Zero's microSD card unless you added Twitter or USB drive functionality.

If you added a USB drive, turn off the camera, remove the drive, and put it into a computer to get your GIFs. Be sure to

replace the USB drive before turning on the camera again or it won't save new GIFs.

If the PIX-E is uploading your GIFs straight to Twitter, go check your tweets!

FINISHING TOUCHES

We created 5 printable wrappers that you can cut out and wrap around the camera to give it that '90s disposable camera feel (Figure L). Download the file *PIX E WRAPPERS.zip* from the project page online, then print out the PDFs on cardstock and follow the included instructions to fold, wrap, and tape them in place. You can preview or grab the individual PDFs at moontowerlabs.com.

You can also experiment with different lenses on the PIX-E. We designed the camera front to accept common cellphone lens kits (Figure M). Have fun!

Watch assembly videos, and fun GIFs made with this camera, and share your build at makezine.com/go/pix-e-gif-camera.

LED Matrix Handbag 2.0

Written by Debra Ansell

Make a dazzling light-up tote that pairs with your phone to display text, animations, or tweets in real-time

Time Required: 2-3 Weekends

Cost: $150-$250

Hep Svadja

DEBRA ANSELL (geekmomprojects.com) studied physics and applied math before becoming a software engineer in the mid-90s. She quit to stay home with her 3 boys after the internet bust, then rediscovered her love of technology as a FIRST Lego League robotics coach. She has been making open source projects ever since.

MATERIALS

FOR THE HANDBAG:

- **Vinyl or leather (about ¾ yard)** Stores specializing in fabrics for the home can be a great source of nice-looking vinyl fabrics.
- **Lining fabric (½ yard)** for handbag interior
- **Accent fabric (½ yard)** for handbag exterior
- **Iron-on fusible interfacing (1 yard)** available at fabric stores
- **Zippers, 16" long (2)** one for the exterior and one for the lining
- **Magnetic snaps, self-adhesive (6 pair)** the super slim kind, such as Amazon #B0033PHCU0
- **Fabric glue**

FOR THE LED MATRIX:

- **Flexible RGB LED strip, APA102C type, 60 LEDs/meter, 2 meters long** such as Adafruit #2240 or 2239
- **Flexible vinyl sheet (½ yard)** for matrix backing, available at many fabric stores. Look for clear vinyl, or a color that matches your handbag colors.
- **Stranded wire, silicone covered, 26 AWG**
- **Arduino-compatible microcontroller** for testing the 5V LED matrix. I used an Arduino Uno (5V) here, but the Adafruit Feather M0 (3.3V) could also be used, with the addition of a logic level converter.

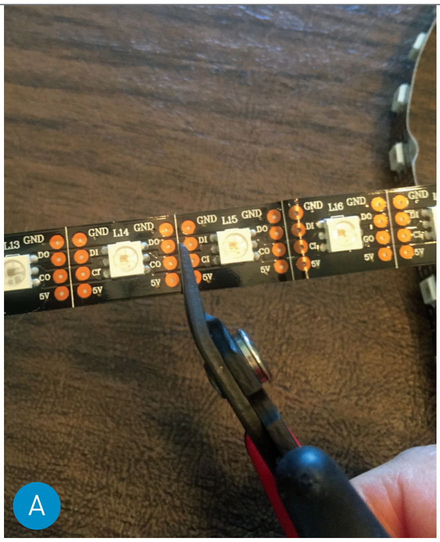
MICROCONTROLLER AND POWER SUPPLY:

- **Adafruit Feather M0 Bluefruit LE microcontroller board** Adafruit Industries #2995, adafruit.com
- **Logic level converter, bi-directional** Adafruit #757
- **2.1mm barrel-jack connector** such as Digi-Key #EJ503A-ND, digikey.com
- **Heat-shrink tubing, 1" or 25mm, or electrical tape**
- **Portable USB charger, 5V** to power the LED matrix
- **USB cable, 2.1mm barrel-jack to standard USB** Adafruit #2697

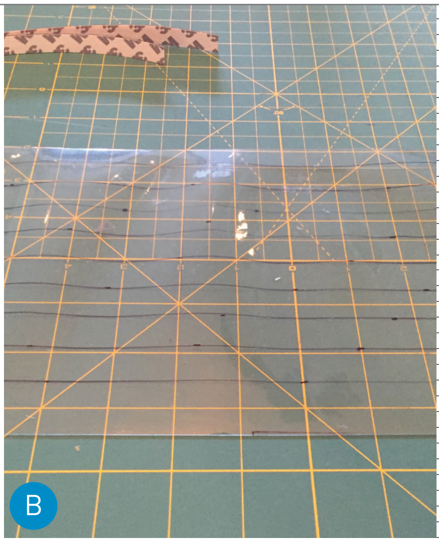
TOOLS

- **Sewing machine with zipper foot attachment**
- **Sewing machine accessories (optional): leather needle, and walking foot attachment** very helpful but not absolutely necessary
- **Scissors**
- **Fine-point marker**
- **Rotary cutter (optional)** with cutting mat and guide ruler; or use fabric scissors
- **Soldering iron and solder**
- **Clothes iron**
- **Binder clips**
- **Computer** for programming the Adafruit Feather microcontroller

Debra Ansell



A



B

HERE'S A DAZZLING, FUN ACCESSORY FOR AN EVENING OUT THAT'S ALSO A GREAT CONVERSATION STARTER — a fabric-topped vinyl handbag with a removable, Bluetooth-connected, full-color LED matrix.

It's bright, beautiful, and interactive, controlled by a smartphone app that sets the display of different low-resolution animations and scrolling text — it can even receive data from Twitter in real-time. And because it's removable, it can easily be used in another wearable project, such as a messenger bag or jacket.

The most challenging part of this project was coming up with a design that organically and seamlessly integrated the LEDs into a handbag that's attractive on its own merits. About a year ago I created my first Twitter purse with a 10×6 matrix but I wanted a better-looking one. So here's version 2.0, with a new, woven matrix cover and a bigger, 14×8 LED matrix. I'm proud of the outcome and really enjoy taking it with me when I go out.

How can a purse talk to Twitter? Visit the project page online at makezine.com/go/led-matrix-twitter-handbag and I'll show you two different ways to program it: one using the Adafruit.io and IFTTT cloud services, the other using a dedicated Raspberry Pi running Mosquitto server and Node-RED.

MAKE YOUR TWEET-POWERED LED HANDBAG

If you have some comfort with a sewing machine, I estimate it'll take you 6–8 hours to make the bag, 8–10 hours to build the electronics, and maybe another 6–8 hours to do all the programming.

BUILD THE ELECTRONICS

1. MAKE THE LED MATRIX

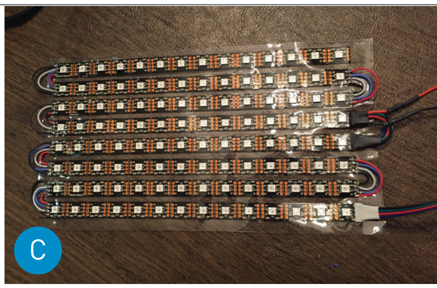
Cut the 2-meter APA102C RGB LED strip into 8 short strips of 14 LEDs apiece, using a wire cutter to cut along the lines between the soldering pads (Figure A). Leave the first 14 connected to the JST "in" connector. (Usually, arrows on these LED strips indicate directionality; data flows in one end and out the other end.)

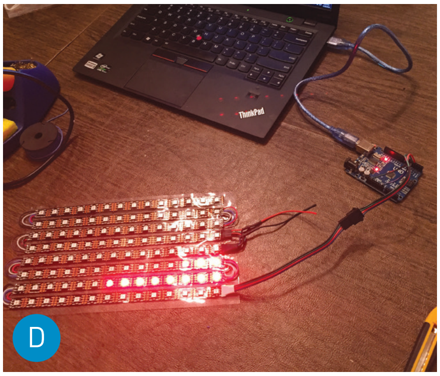
There will be 8 LEDs left over, which you can discard or use for another project — but save the JST connector from the "out" end. You'll use it later in the project.

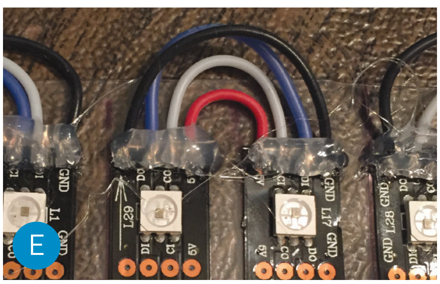
> **TIP:** Most LED strips have a join every ½ meter. The spacing at this join is different and can ruin the symmetry of the matrix, so cut your strips from *between* the joins.

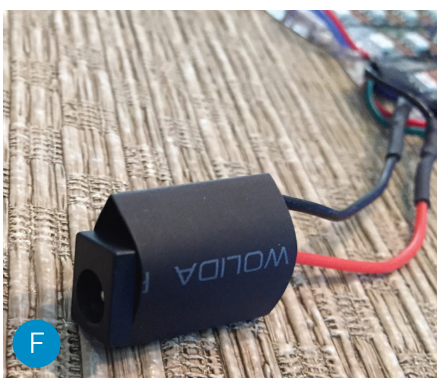
Cut a piece of clear vinyl backing about 28cm×14cm. Larger is fine; you can always trim it later. Leave a margin of vinyl at least 1cm wide around the LED strips. Use a fine-point marker to draw 8 parallel lines spaced 1.6cm apart (Figure B).

Line up the LED strips on the vinyl, orienting them in a back-and-forth pattern so that current and data can flow out the end of one strip, into the beginning of the next. Be sure the strip that's still connected to the "in" JST connector is the first row of your matrix. Peel the backing off each LED strip and press it onto the vinyl.


C


D


E


F

2. SOLDER AND TEST THE MATRIX

Cut the 26 AWG stranded wire into short lengths to connect each LED strip to its neighbors. Tin the pads on the end of each strip, and the ends of each wire, before soldering. Be careful that the stranded wire doesn't spread out and make contact with a neighboring solder pad.

Somewhere in the middle of the matrix, solder on an extra wire each for power and ground to the end of one of the strips and leave them loose (Figure C). These will be used to supply power to the matrix.

Now test your connections. Connect your matrix to an Arduino Uno (5V, GND, pin 4 for data, pin 5 for clock) and run Adafruit's test sketch from github.com/adafruit/Adafruit_DotStar/blob/master/examples/strandtest/strandtest.ino to make sure all the LEDs light up correctly (Figure D).

Once tested, use a hot glue gun to cover the solder joints (Figure E). This provides strain relief for the wires and also prevents any stray strands from coming loose and causing a short.

3. ATTACH THE POWER JACK

Solder a 2.1mm barrel-jack connector to the power and ground wires you left loose on the last step. Use shrink tubing around the connector to provide strain relief on the wires (Figure F). This barrel jack port can now be connected to any 5V regulated supply to power the LED matrix. A 5V lipstick phone charger in conjunction with a USB-to-barrel-jack cable works well as a portable power source.

4. CONNECT THE MICROCONTROLLER

Solder connections to the Adafruit Feather M0 Bluefruit board and the logic converter as shown in Figure G. Solder the 4 wires leading off-board to the mating JST connector you saved in Step 1, so you can plug and unplug your matrix.

5. MAKE AN ENCLOSURE

Find a box or make a protective housing for the electronics. I 3D-printed a box with a hole to allow a micro-USB connection to the Adafruit Feather. You can download the design files for the box from github.com/geekmomprojects/woven-led-handbag.

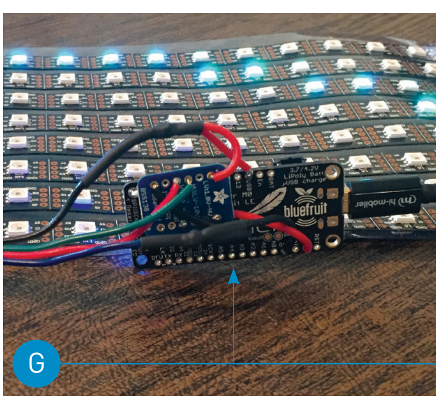
SEW A FABRIC-TOPPED HANDBAG

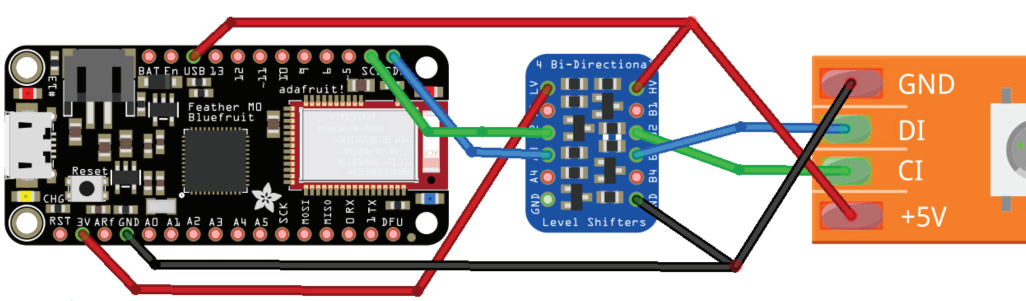
6. WEAVE THE LED MATRIX COVER

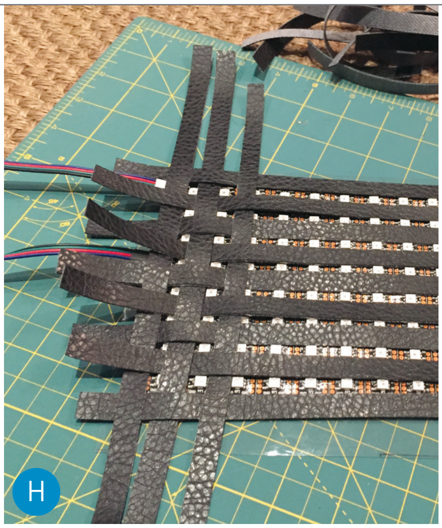
The matrix cover is made by weaving thin strips of vinyl (or leather) between the LEDs. The spacing between LEDs is 1.1cm, so you'll need 24 strips 1.1cm wide. Mark the spacing on the wrong side of the vinyl and cut the strips with a rotary cutter and ruler (recommended), or with fabric scissors.

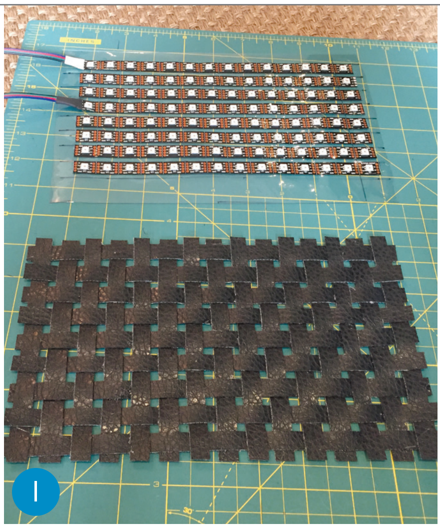
Lay out the horizontal vinyl strips between the LED rows, with one strip above and below the top and bottom rows as well. Then weave vertical strips between the horizontal ones so that only the LEDs show through the holes in the woven grid. It's easiest to do the weaving at the end of the horizontal strips, then slide the vertical strips into position (Figure H). Be sure there are strips all around the perimeter.

Once the grid is complete, use a toothpick to apply small dabs of fabric glue to secure the vinyl strips in place. Start with a corner of the grid, and work your way around the perimeter, very lightly tacking all the overlapping strips to each other (*not* to the LED matrix). Once the perimeter is glued, tack enough of the strips in the center of the grid so that they won't slide around when the grid is lifted. Let dry at least 1 hour, then lift it off the LED matrix and trim the excess fringe (Figure I).


G







7. CUT THE HANDBAG PIECES

While the glue on the grid is drying, cut the pieces for the handbag body as follows: vinyl 16×9½" (2), accent fabric 16"×5½" (2), lining fabric 16"×14" (2), and 1 pocket made from the lining fabric, sized as you desire.

8. SEW THE GRID TO THE FRONT PANEL

Choose one of the vinyl pieces to be the front of the handbag. Then cut a window just large enough to display all the LEDs in the matrix. The outer perimeter of the LEDs in my matrix is about 21.8cm wide by 11.7cm high, but you should measure yours in case there are variations. Draw a rectangular hole, matching your perimeter, about 1½" down from the top and centered, and then cut it out (Figure J).

On the wrong side of the vinyl, draw a light line of fabric glue around the window and glue your vinyl matrix cover into place. Allow the glue to dry at least 1 hour and then sew around the perimeter of the hole with a ⅛" seam to hold the matrix cover in place (Figure K).

TIP: A leather needle and walking foot attachment for your sewing machine will make sewing several layers of vinyl together much easier.

9. ADD THE MAGNETIC SNAPS

Magnetic snaps let you remove your LED matrix and use it in other projects! Align your LED matrix with the vinyl grid, then peel the backing off the magnetic snaps and stick them around the outside edges of the matrix, and in corresponding places on the vinyl (Figure L). Six should be enough. Be sure to use magnets of opposite polarity so they'll attract each other. If you don't trust the adhesive, you can reinforce them with tape or fabric glue. I used duct tape.

10. ADD THE ACCENT FABRIC

Take your accent fabric pieces and cut 2 smaller pieces of fusible backing (interfacing) to leave a perimeter of about ½" all around (Figure M). Iron on the interfacing, following the manufacturer's directions. Then, using binder clips (since pins would make holes in the vinyl), clip the accent fabric to the vinyl with the right sides together (Figure N).

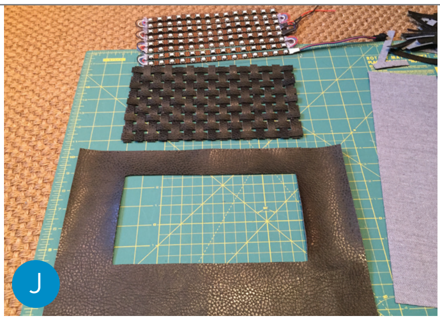
Sew the accent fabric to the front vinyl piece along the top edge, using a ½" seam allowance. Press open and flatten the seam allowance with your fingers (don't attempt to iron the vinyl!), then topstitch just above and below the seam so it lays flat (Figure O). Repeat on the back vinyl piece.

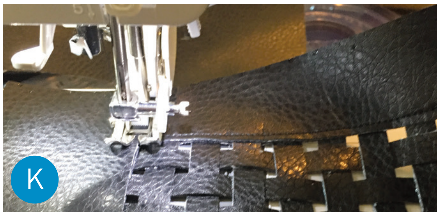
11. PREPARE THE LINING PIECES.

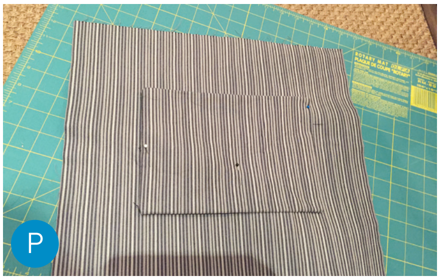
First cut 2 pieces of fusible interfacing about ½" smaller than the lining pieces and fuse the interfacing to the lining.

On the back piece of lining sew in a slip pocket (Figure P), or any other kind of pocket you'd like.

On the front piece of the lining, cut off a 3" horizontal strip, and sew in a zipper that goes all the way across (Figure Q). This will provide access to the electronics. Topstitch the zipper on both sides so it lies flat. Then, if the height of the front lining piece has become more than 14" due to the zipper, trim excess fabric from the bottom to make it 14" again.







12. SEW THE TOP ZIPPER

The ends of the exterior zipper get covered with small vinyl patches to give a more finished look. Cut 2 small 2"×3" pieces of vinyl. Fold the 2" edges under by about ½", then clip them to the ends of the zipper so that the exposed length of zipper is slightly smaller than the width of the handbag (Figure R). Then sew across the folded vinyl pieces, about ¼" from the fold, to attach them to the zipper (Figure S).

Next, the zipper gets attached to the front of the bag. On a flat surface, lay the front of the handbag out right side up. Lay the zipper along the top edge with the right side down. The top edge of the zipper should just line up with the top edge of the accent fabric. Then place the front lining piece over the zipper facing right side down. Clip the top edge of all 3 elements together. Sew along the length of the zipper using a zipper foot attachment. After sewing, flip the wrong sides of the bag and lining together.

Repeat the same process to attach the zipper to the back pieces of the bag and lining. After sewing, you should be able to open up the bag with the front vinyl and the front lining pieces wrong sides together and the back vinyl and back lining pieces wrong sides together (Figure T).

Next, topstitch along the accent fabric on both sides of the zipper.

13. SEW THE OUTSIDE EDGE

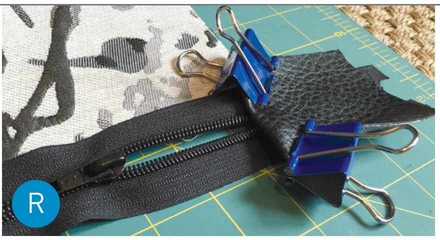
Open up the bag so the right sides of the outside face together and the right sides of the lining face together. Pin around the perimeter of the bag, doing your best to line up the seams where the accent fabric meets the vinyl (Figure U).

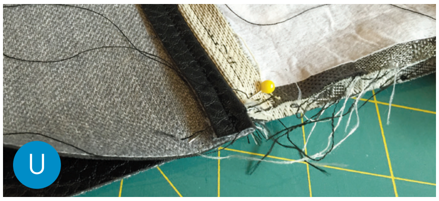
IMPORTANT: Before sewing around the bag, unzip both zippers halfway. If you don't, it will be almost impossible to invert the bag later.

Then sew all the way around the outside edge of the bag with a ½" seam allowance. As you traverse the perimeter, stop sewing at the exterior zipper then start up again on the far side.

14. SQUARE THE CORNERS

Before inverting the bag, square both the vinyl corners and the lining corners as follows: Take each corner, flatten it





out, and line up the seams on top of each other. Measure and mark a line 3" long, perpendicular to the top seam. Sew across the marked line, being sure to keep the seams as aligned with each other as possible.

Once sewn, cut off the excess corner fabric (Figure V). The sewn bag should now look like Figure W.

15. INVERT THE BAG

Pull the whole bag through the open zipper in the lining (Figure X). Then push the lining into the handbag through the main zipper (Figure Y).

16. ADD STRAPS (OPTIONAL)

You can find webbing straps in many fabric stores. Cut them to whatever length you will find most functional.

To sew your own, use 3"-wide vinyl strips, placed right sides together then sewn lengthwise with a ½" seam. Open up the straps and fold the wrong sides together, tucking a ½" seam from the unsewn side of each piece between the 2 strips. Secure the straps along their entire length with clips, and sew along the length with a ¼" seam, pressing the straps flat and removing the binder clips as you go. Then sew a matching seam ¼" from the opposite side. If you wish, you can add additional decorative seams along the length of the straps about ⅛" from the edge.

Tuck the ends of the straps under, between the strap and the handbag, before sewing the straps to the bag (Figure Z). Here a walking foot attachment and leather needle will be very helpful.

PROGRAM YOUR LED MATRIX HANDBAG

17. PROGRAM THE MICROCONTROLLER

The Arduino code for your matrix drives several different animations and also receives text input via UART Bluetooth communication. Download the source code at github.com/geekmomprojects/woven-led-handbag. Open it using the Arduino IDE software on your computer, then upload it to your Feather M0 Bluefruit board. The code requires installation of the FastLED (fastled.io) animation library to control the LED matrix and the Adafruit BLE and Bluefruit libraries (learn.adafruit.com/adafruit-feather-32u4-bluefruit-le/installing-ble-

library) to allow BLE control of the display using the Adafruit Bluefruit app on your phone or tablet. If you're not sure how to install a library into the Arduino IDE, visit arduino.cc/en/Guide/Libraries.

IMPORTANT: Any time you connect the Feather to your computer for programming, you should have the matrix connected to a separate, regulated 5V power source. Otherwise the matrix will try to pull all the power it needs through the Feather, which may damage the board.

18. TEST IT

Once the code is loaded onto the Feather board, connect the LED matrix to a 5V power source to start it running. After about a 10-second delay, the first animation will start running!

19. CONTROL IT WITH BLUEFRUIT APP

To change animations or scroll text, just install the Adafruit Bluefruit App for iOS or Android on your mobile phone (Figure AA). When you start the app, you'll see a list of available BLE devices. Connect to the Feather board, shown as the "Bluefruit LE" device. A page with information about the Feather BLE will appear (Figure BB). Select the UART icon from the bottom of the screen to communicate with the board via text commands and messages (Figure CC).

SCROLL IT LOUD, SCROLL IT PROUD

Now you can send typed commands to your handbag. Any text that's not preceded by the **!** symbol will be displayed on the matrix as written, and scrolled across the handbag display 3 times. Try it out: Type a few words into the box on the bottom of the screen and then hit Send.

Any text that is preceded by a **!** symbol is interpreted as a command. The following commands will change the handbag display:

`!next` or **`!n`** — moves to the next animation in the sequence

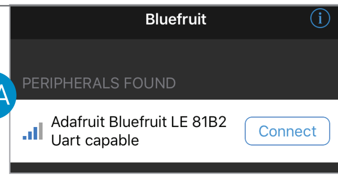
`!bxxx` — where **xxx** is a 3-digit number, sets the brightness level from 0–255

`!pal` — selects a different color palette for the animations.

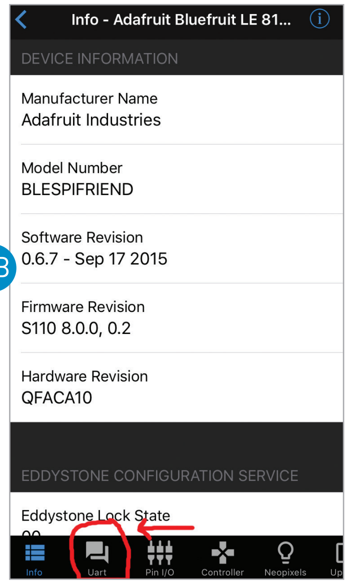
Palettes are chosen from a pre-set sequence defined in the code.

Now take your new LED handbag for a night out!

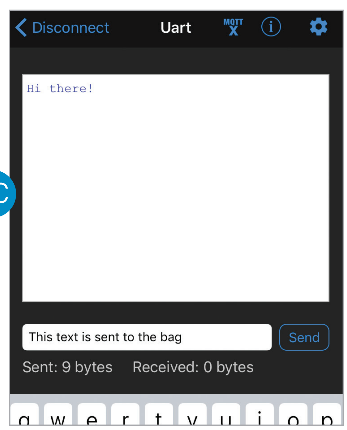
AA



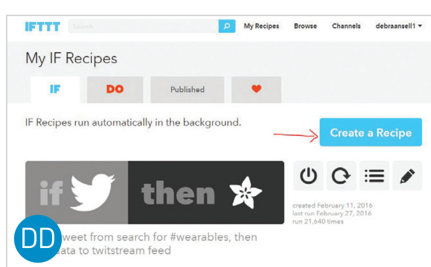
BB



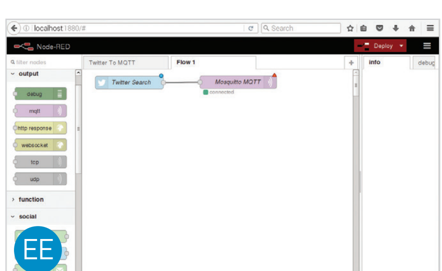
CC



DD



EE



CONNECTING YOUR HANDBAG TO TWITTER

If you like, you can automatically send Twitter data to the handbag display through the Bluefruit App's connection to an MQTT feed. MQTT is a lightweight messaging protocol that's often used in IoT communication. On the project page online, I'll show you 2 different workflows for setting up a Twitter-handbag connection using MQTT:

➡ **METHOD 1** uses 2 online services — ***IFTTT*** and ***Adafruit.io*** — to route the data (Figure DD). It's the easier method to set up, and runs automatically online, but gives you less control over the timing of the data.

➡ **METHOD 2** requires setting up your own ***Mosquitto (MQTT) server*** and running ***Node-RED***, a graphical tool that sends data from one application to another (Figure EE). It's more work intensive, but gives you better control over data flow and a faster response to new tweets. I use a dedicated Raspberry Pi for this, but you could use any computer with a constant internet connection.

Learn how to do it either way, at makezine.com/go/LED-matrix-twitter-handbag.

GOING FURTHER

Since the LED matrix is removable, you can easily create other types of handbags or wearable projects to use it in. I made an evening bag version that I can take with me on a dressy evening out. It's fun to coordinate the color palette of the animations with what I'm wearing at the time.

You can also easily customize the data displayed on the bag by changing what information is sent to the MQTT feed. Instead of tweets, it could display a news feed or weather data.

You could make a second bag for a friend and synchronize the displays. Use your imagination!

Watch the LED Matrix Handbag animations, set up a Raspberry Pi as your own MQTT server, and share your build on the project page at makezine.com/go/LED-matrix-twitter-handbag.

Willis Carrier: **One Super Chill Dude**

Stay frosty with a cheap, effective DIY air conditioner based on his 1902 evaporative-cooling original

Written by William Gurstelle

WILLIAM GURSTELLE's new book series *Remaking History*, based on this magazine column, is available in the Maker Shed, makershed.com.

PRIOR TO THE WIDESPREAD ADOPTION OF MODERN AIR CONDITIONING IN THE FIRST HALF OF THE 20TH CENTURY, life was far different than it is now. Summer was often a time of heat-induced doldrums, when stores and factories simply shut down during heat waves. When the temperature and humidity climbed, the only recourse for many people was to spend large portions of the workday lying about on patios or fire escapes. And the farther south you traveled, the more unbearable the heat became; places like Florida and Texas were nearly uninhabitable, and few businesses would consider permanently locating in spots so unconducive to their workers.

Washington, D.C., was so hot and miserable during the dog days of summer that President Woodrow Wilson couldn't bear working inside the White House. Instead, he set up a large tent in the Rose Garden so he could work in a place slightly more tolerable than the stifling Oval Office.

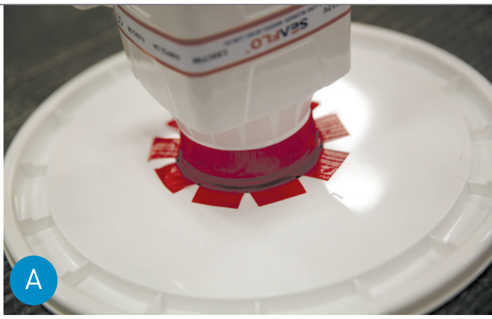
But that sticky situation would be changed by a New York engineer named Willis Carrier, who designed and built the first air-conditioning system in 1902. Carrier knew a great deal about thermodynamics and in particular, he realized that when fluids changed from a liquid state to a vapor state, they did so by absorbing heat from their surroundings. This process, which is the basis for modern air conditioning systems, makes use of the concepts of *phase changes* and *latent heat*.

In a nutshell, when a liquid evaporates (a phase change) into surrounding air, it cools any object or a liquid in contact with it. The amount of cooling varies by the properties of a liquid's latent heat, which is a physical property associated with the amount of heat that is needed to evaporate the liquid. If more liquid is evaporated, then more heat is pulled from the surroundings, and the amount of cooling provided by the system is greater.

Carrier's first successful cooling project was a New York City printing plant. The results were so good that soon other factories clamored for a way to cool and dehumidify the air, a process that eventually came to be called "air conditioning." In 1915, he started the Carrier Engineering Company and before long the company was designing and installing air conditioning systems for office buildings, theaters, and factories across America.

In 1930, Carrier installed a modern air conditioning system that cooled the West Wing

Hep Svadja



of the White House. No longer would presidents conduct the nation's business in tents on the White House lawn.

VAPOR COMPRESSION VS. WATER EVAPORATION

It's not hard to build a DIY air-cooling system that works on principles of latent heat and evaporation. Basically, in our DIY cooler, water is mixed with air and the water is evaporated. As it does so, a great deal of energy in the airstream changes from heat you can sense, called *sensible heat*, to heat you can't, called *latent energy*. The air exiting the blower becomes cooler and more humid.

It's important to understand that an evaporative cooling machine is not the same as a mechanical vapor-compression refrigeration machine. It is true, however, that both processes make use of the concepts of evaporation and latent heat.

In a mechanical vapor-compression cycle, after the refrigerant evaporates inside the system's evaporator coils, the refrigerant gas is re-compressed and cooled, which allows it to return to its liquid state. The refrigerant is reused and never enters the atmosphere, which is a good thing because refrigerants are both expensive and polluting.

But an evaporation system is based on the latent heat of water, and the evaporated water is introduced into the space along with the air. So now the air is not only cooled but humidified — which explains why evaporative coolers are best suited for dry, hot environments.

BUILD A DIY AIR CONDITIONING SYSTEM

Also known as a *swamp cooler*, this device is perfect for hot, dry days. (Its cooling potential decreases as humidity increases.)

1. Use heavy-duty scissors, or a cutoff wheel in a handheld rotary tool, such as a Dremel, to make three 9"×4½" horizontal openings near the top of the bucket, equally spaced.
2. Use a compass to draw a 3¼" circle in the center of the lid for the plastic pail. Use heavy-duty scissors to cut out the circle.
3. Insert the round inlet duct of your fan into the hole in the lid, from the top. Fix into place using waterproof tape (Figure A).
4. Connect the flexible metal duct to the outlet of the fan using waterproof tape.
5. Insert the bolt into one end of the vinyl tubing. Use an awl to poke a hole every ½" in the tubing (Figure B), extending back 30" from the bolt.
6. Insert the cooler pad into the bucket. If necessary, trim it with scissors so it fits neatly around the bucket walls (Figure C).
7. Drill a pair of ⅛" holes in the bucket, one slightly above the other, just above the top edge of the cooler pad. Repeat every 90° around the bucket, for 4 total pairs of holes.
8. Set the plastic tubing atop the cooler pad, and secure it in place by inserting 4 pieces of 4"-long craft wire through the ⅛" holes and tying them off. Take care to keep the tubing section with the holes centered above the fabric.
9. Attach the submersible pump's outlet to the plastic tubing and place the pump in the bottom of the bucket. Extend the pump wiring outside the bucket by running it under the fabric and out one of the 9"×4½" openings.
10. Fill the bucket with 5" of water.
11. Place the lid with the fan on top of the bucket. Extend or direct the duct as desired. Connect the pump and fan to the battery (Figure D).

Enjoy your cool air! Depending on weather conditions, you can produce an airstream with 15°F of cooling, or even more. If you use a solar panel to charge the battery, you'll have made one of the lowest-cost and greenest cooling devices possible.

Have you built a bucket air conditioner for Burning Man or summer campouts? Share tips and see more photos at makezine.com/go/remaking-evaporative-cooler.

William Gurstelle, Hep Svadja

Time Required: An Afternoon
Cost: $50–$60

MATERIALS

- **Plastic pail with lid, 5gal** about 12" diameter by 14" high
- **Fan or blower, 3" diameter, 12V, 130 CFM** We used an in-line marine bilge blower, Amazon #B00F7ANK7S.
- **Submersible fountain pump, 12V** with ⅜" exit port
- **Vinyl tubing, ⅜" inner diameter, 5' length**
- **Flexible aluminum duct, 3" diameter** bend-and-stay type
- **Evaporative cooler pad, polyester, 30"×13"×1"** such as Dura-Cool brand
- **Battery, 12V** Our bilge fan draws 2.5A at 12V, so a rechargeable 12V battery rated at 7 amp-hours (7Ah) will last for nearly 3 hours.
- **Craft wire**
- **Waterproof tape, 2" wide**
- **Bolt, ⅜", ¾" long**

TOOLS

- **Heavy-duty scissors** or cutoff wheel in handheld rotary tool (e.g. Dremel)
- **Drawing compass**
- **Awl**
- **Drill and ⅛" bit**

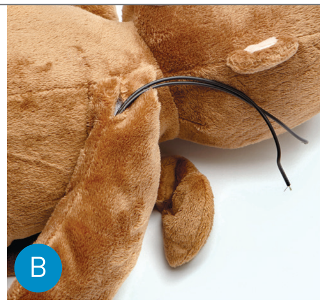
Shock the Monkey

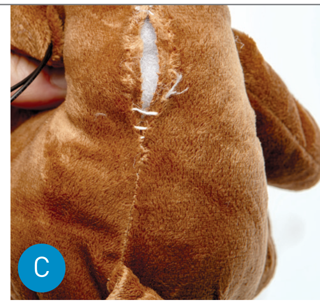
Written by Emily Coker and Kelli Townley

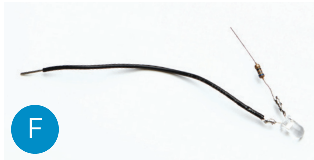
Give your stuffie a beating heart with a flashing LED and embroidery thread

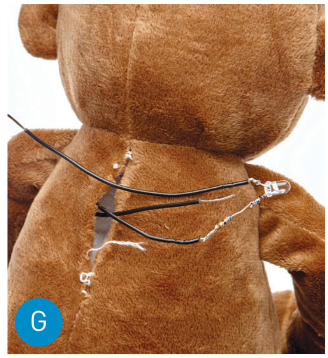
Rory Earnshaw

This project is excerpted from *Make It Glow: LED Projects for the Whole Family*. Find it at the Maker Shed (makershed.com) and fine bookstores.









YOU KNOW YOUR STUFFIES LOVE YOU — ALMOST AS MUCH AS YOU LOVE THEM. Here's a way to show it! A flashing LED makes a beating heart on the inside, and a few stitches with embroidery thread show the love on the outside.

PLACE THE SWITCH

1. With the scissors or craft knife, open a small hole in your stuffie's arm. Cut along a seam, if you can, so the arm will be easier to sew back together (Figure A). Then feed the head of the tactile switch down into the arm so the leads are sticking out (Figure B). (If you need to remove a bit of stuffing to get the switch in, just stuff it back in when you're done.)

2. Carefully snip another hole in the back of your stuffie (again, on a seam, if possible, Figure C). Using transparent tape, wrap together the ends of the leads sticking out your stuffie's arm and thread them, like a shoelace, through the stuffie and out the back (Figure D). Set your stuffie aside.

WIRE THE LED INTO THE CIRCUIT

3. Mark the long, positive (+) leg of your LED with the black marker. Then gently bend the legs of the LED apart. With the wire strippers, cut a 4" length of hookup wire and strip about ½" of insulation off both ends. Using the soldering iron, solder one end to the negative (–) leg of your LED (Figure E).

TIP To make soldering easier, use your third-hand tool to hold the pieces in place, and hold the wires steady with your needlenose pliers when you strip them.

4. With your wire strippers, snip the positive leg of the LED to shorten it (this will keep the wire from flexing and breaking). Twist one end of the resistor (either end will work) onto the shortened LED leg, and solder in place (Figure F). As noted in the tip box, use the third-hand tool to help hold the pieces.

5. Using wire strippers, snip one lead of your switch (it doesn't matter which one) to shorten it about 1". Then strip that wire, twist it to the resistor's free side, and solder in place (Figure G).

6. Wrap electrical tape around both sides of the LED, covering the resistor and all solder points. Use a hot glue gun to cover the connections on both sides of the LED with glue to prevent breaks and short circuits (Figure H on the following page).

7. Test it! Hold a battery between the 2 wires to be sure your LED lights up.

HEART YOUR STUFFIE

8. Wiggle your fingers inside your stuffie and figure out where its heart (your red LED) should go. Use a pencil or piece of chalk to mark the spot inside, so the mark doesn't show through to the front. Using the hot glue gun, put a blob of glue there, wait until it's tacky, and then set the LED in place, right up against your stuffie's chest. Hold until dry (Figure I on the following page).

NOTE If you need more hands to hold the stuffie while placing the LED, ask a friend to help!

9. If you want, as a guide while sewing, use a pencil to draw a heart shape on the outside of

EMILY COKER is a circuit board whisperer, sorcerer of robots, lord of the workshop, and author. She is currently a workshop technician at Google X.

KELLI TOWNLEY is a lifelong creator and tinkerer who has worked in video games, VFX/animation, education, and most recently, virtual reality.

Time Required:
2–3 Hours
Cost:
$20–$30

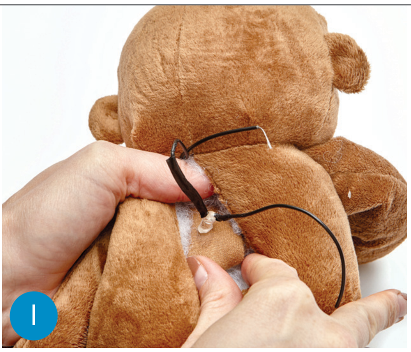
MATERIALS

- **Stuffed animal** that you'd like to hack
- **Tactile on/off switch** with leads
- **Transparent tape**
- **5mm slow-flashing red LED**
- **Hookup wire**
- **Resistor, 68-ohm, ¼-watt**
- **Electrical tape**
- **Red embroidery thread**
- **Sewing thread matching the color of your stuffie**
- **Coin cell battery, 3V, CR2032 type**
- **Coin cell battery holder, CR2032 type**
- **Scraps of felt: 1"×½" (1) and 1"×1½" (1)** any color
- **Sew-on snap**

TOOLS

- **Scissors** Sewing scissors work best, if you have them.
- **Craft knife**
- **Black marker**
- **Wire strippers**
- **Soldering iron and solder**
- **Third-hand tool**
- **Needlenose pliers**
- **Hot glue gun and glue sticks**
- **Pencil or chalk**
- **Sewing needles**


H


I


J

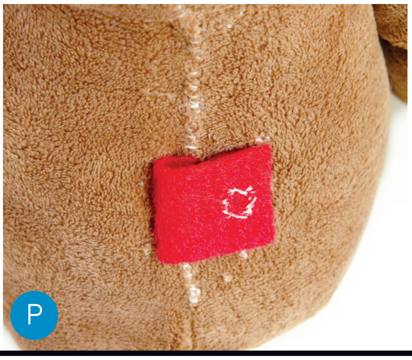

K


L


M


N


O


P


Q

your stuffie, around the LED. Then, thread a needle with doubled-up red embroidery thread. Start from the inside so you can hide the knots, and carefully stitch a heart around the LED on the outside of your stuffie's chest (Figure J).

10. Use sewing thread and a needle to sew up your stuffie's open arm. Knot and cut the thread.

11. Use sewing thread and a needle to sew up your stuffie's back. Sew around the wires, leaving them sticking out, and leave an opening of about 1" so you'll always have access to the battery (Figure K). Be sure to knot and secure the threads above and below the opening.

WIRE THE BATTERY HOLDER INTO THE CIRCUIT

12. Hold or temporarily tape the battery between the 2 wires. Does the LED light up on the front of your stuffie? If not, switch the battery around. When the LED lights up, note which wire is against the positive (+) side of the battery and which is against the negative (–) side. That will tell you how to orient the battery holder (Figure L).

13. Twist the wires into the battery holder, top and bottom, connecting positive (+) to positive (+) and negative (–) to negative (–). (On our monkey, the top wire was the positive wire.) Slip in the battery to be sure the LED is working. When everything is oriented correctly, reinforce both joints. Using the soldering iron, solder the wires in place, and using the hot glue gun, cover them with hot glue. Then carefully stuff the battery and battery holder into the opening.

ADD BATTERY ACCESS AND A SNAPPY FINISH

14. Using a needle and sewing thread, sew one half of the snap in the center of the small piece of felt, and the other half of the snap centered at the edge of the large piece of felt (Figure M).

15. Using the hot glue gun, glue the small piece of felt (snap side out) on one side of the 1" opening (Figure N). Then, making sure the 2 halves of the snap align, glue the edge without the snap of the larger piece of felt to the other side of the opening (Figure O).

Fold to snap the 2 halves together (Figure P).

Press the tactile switch in your stuffie's arm to turn on its beating heart (Figure Q), and give it a heartfelt hug. Surgery successful!

Share your stuffie hacks and mods at makezine.com/go/beating-heart-stuffie.

Rory Earnshaw

Necktie Glasses Case

Fashion a stylish carrying case from a great old tie

Written by Diane Gilleland

DIANE GILLELAND (craftypod.com) is a big crafty geek, and author of *Kanzashi in Bloom* and *All Points Patchwork*. She loves her some English paper piecing and plastic canvas.

Time Required: 30–45 Minutes

Cost: $0–$5

MATERIALS

- **Necktie** not too skinny
- **Thread** in a coordinating color
- **Velcro dots, ½" (1 set)**
- **Fabric glue**
- **Button, 1"** or larger

TOOLS

- **Sewing needle**
- **Scissors**
- **Seam ripper**

A THRIFT STORE NECKTIE IS THE PERFECT RAW MATERIAL FOR A CARRYING CASE: It's elegant and comes with its own padding to protect your valuables. Plus, you need only a few minutes and a little hand-sewing to whip one up. Here's how to make a case for eyeglasses; they're also great for crochet hooks, scissors, business cards, and pens.

1. MEASURE AND CUT THE TIE

Lay the necktie flat, back facing up. Place your folded glasses on top, lined up with the widest part of the tie. Fold the rest of the tie over the glasses, and cut it so it extends 1" beyond the top of the glasses.

2. FINISH THE CUT EDGE

Fold 1" of the cut edge toward the back of the tie and whipstitch it in place. If the tie's internal padding sticks out, turn that under as you stitch.

3. OPEN UP THE INSIDE

Most ties are made with a hand-sewn center seam — you'll need to remove some of this seam in order to fit things inside your case. Use a seam ripper to remove the stitching. You only need to remove enough stitching to accommodate your glasses.

If your tie has a label, remove this too.

4. SEW THE SIDE SEAMS

Set your glasses aside and fold the tie as shown. Pin the 2 layers together and hand-sew them along both sides with a hemstitch or a tiny whipstitch. Stitch only through the topmost layers of fabric — you don't want to sew the front and back of your case together!

5. ADD A CLOSURE

Fold the tip of the tie over to make a flap. Sew the button on the outside of this flap.

Glue velcro dots under the button to hold your case closed. Once the glue is dry, your glasses are good to go.

Share your upcycled necktie projects at makezine.com/projects/necktie-glasses-case.

Hep Svadja, Diane Gilleland

The Dishonest Decider

Time Required: A Weekend
Cost: $20–$30
Written by Charles Platt

Build a random yes-no circuit you can secretly control with a hidden switch

PROBABLY YOU'VE SEEN THOSE LITTLE "EXECUTIVE DECISION MAKERS" that illuminate an LED to assist you in making a yes-or-no choice. Personally, I don't want to reach decisions on a random basis — but what if the output could be faked? That suggests some interesting possibilities.

Suppose your friends want to eat at a cheap diner where the greasy, spicy food gives you heartburn. "I suppose we could go there," you say, "but let's see what my decision maker thinks." You pull out your handy gadget, slide the secret switch to the "no" option, press the button, and it gives you exactly the output that you wanted.

A toy of this type has never been marketed, so far as I can tell. But now, with a handful of parts, you can make your own. I call it the Dishonest Decider.

BASIC PRINCIPLE

In Figure A, a 7555 timer (using less power than a 555) is wired in bistable mode. When the Run button is pressed, the timer's Reset pin is pulled low, forcing its output low, which grounds three LEDs representing Yes, Maybe, and No. The LEDs are flashed in sequence by a decade counter driven by a free-running 7555 at the bottom of the circuit. The LEDs keep blinking so long as you hold down the Run button.

Now for the interesting part. When you

Hep Svadja

release the Run button, the LEDs stop flashing — but not immediately. The Disable pin of the counter is controlled by the output from an AND gate. When the Run button is released, a pull-up resistor makes one input to the AND gate go high. The other input only goes high depending on the position of a concealed rotary switch, which has selected the LED that you want. The counter only stops when that LED is lit.

To use the Dishonest Decider, you secretly set the switch, then press and release the Run button — or let someone else press and release it. Either way, the Decider doesn't stop until your choice is selected.

A Reset button pulls the input of the bistable timer low, so its output goes high. Because the LEDs are grounded into this output, it stops them from glowing. You can now readjust the position of the rotary switch, if you wish, while the LEDs remain dark. Note that the circuit still uses power in this mode, and a separate on-off switch is necessary to avoid draining the battery.

What if someone suspects that the Decider is — well, somewhat dishonest? No problem! Simply turn the rotary switch to its fourth position, which holds the left-hand input of the AND gate constantly high. Now the counter will stop as soon as the Run button is released, and anyone can test the Decider repeatedly, because its output is genuinely random in this mode.

A schematic is shown in Figure B. This includes everything except the power supply, which can be a 5V AC/DC adapter or a 9-volt battery with an LM7805 voltage regulator.

CONSTRUCTION

To hide the rotary switch while allowing you to turn it surreptitiously, I mounted it inside a 3" PVC coupling — a piece of plumbing that you should find in any large hardware store. The Dishonest Decider is shown completed in Figure C, with its base visible in Figure D. When you rotate the base, this turns the internal switch. Three wooden feet make the base easier to grip.

Fabricating the Decider must be done accurately. First check the internal diameter of your coupling, as in Figure E. Punch 2 small holes spaced half of this distance apart in a piece of cardboard, and use it to draw a circle on a piece of ¼" plywood, as in Figure F. To cut around the circle, the

Charles Platt

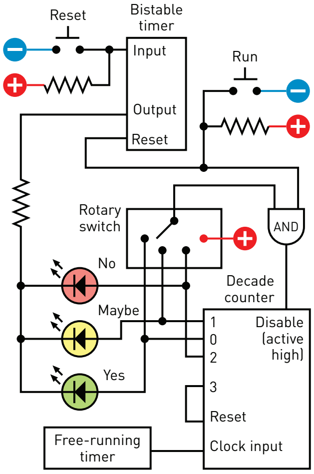
MATERIALS

- **» Timer IC chips, 7555 type (2)**
- **» Decade counter chip, 74HC4017 type**
- **» AND chip, quad 2-input, 74HC08 type**
- **» Switch, on-off toggle, subminiature** SPST or SPDT
- **» Switches, pushbutton, normally open (2)**
- **» LEDs with internal resistors (recommended): red (1), amber (1), and green (1)** I used Chicago Miniature Lighting #4302F1-5V, #4302F3-5V, and #4302F5-5V. You can also use ordinary LEDs but you'll have to add a 330Ω resistor.
- **» AC/DC adapter, 5V** You can substitute a 9V battery with LM7805 voltage regulator and 0.33μF and 0.1μF smoothing capacitors (ceramic).
- **» Capacitors, ceramic: 0.01μF (1) and 1μF (1)** for timing
- **» Resistors, 10kΩ (4)** for timing and pull-up resistors
- **» Rotary switch, 4-position, 1-pole, single-deck, with matching push-on knob** 2-pole or 3-pole switches are also OK. Knob should be 1" diameter or larger with a flat surface that's easy to drill.
- **» Ribbon cable, 12" long (optional) and headers (optional)** for off-board connections
- **» Perf board, unplated, 0.1" hole spacing, 4" square**
- **» Hookup wire**
- **» PVC coupling for 3"-diameter waste pipe** for enclosure
- **» Plywood, ¼" thick, 12"×12" minimum**
- **» Round wood dowel ⅝"**
- **» Sheet metal screws, flat-head: ⅝" #4 (12) and ½" #8 (2)**
- **» Velcro tape, double-sided, ½" wide, 6" length** to secure battery
- **» Transparent epoxy glue and hardener**
- **» Sandpaper, 100 grit**
- **» Polyurethane wood finish**

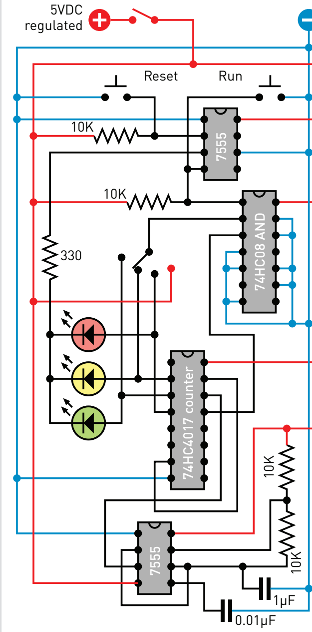
TOOLS

- **» Soldering iron**
- **» Drill and twist bits**
- **» Coping saw or jigsaw** to cut plywood
- **» Hacksaw** to cut perf board
- **» Brush or rag**

CHARLES PLATT is the author of *Make: Electronics*, an introductory guide for all ages, its sequel *Make: More Electronics*, and the 3-volume *Encyclopedia of Electronic Components*. His new book, *Make: Tools*, is available now. makershed.com/platt



A Block diagram showing the principle of the thing.



B Schematic layout suitable for breadboarding.

C

D

E

F

G

H

I

J

cheapest tool is a coping saw, as shown in Figure G.

At this point I can't resist mentioning my new book, *Make: Tools*. This is a comprehensive guide to workshop tools, with a lot of information about saws and plywood — and plastics, too. It's especially useful when you want to build enclosures for electronics projects.

The Dishonest Decider needs a 1-pole, 4-position rotary switch, but you can use a 2-pole or 3-pole switch and ignore the extra poles and contacts. You also need a push-on knob that matches the switch. I found these parts cheaply by searching for "rotary switch" on eBay.

A typical plastic knob is shown in Figure H. Drill 2 holes in it, as in Figure I, then invert the knob and drive 2 screws up into it through matching holes in the wooden disc, as in Figure J. The holes in the disc must be slightly larger than the thread diameter of the screws, while the holes in the knob must be smaller, so the screws will bite into it. The knob must be precisely centered.

You can make feet for the Decider by cutting slices of 5/8" round dowel, applying epoxy, and clamping them as in Figure K. It's easiest to round the end of the dowel with sandpaper before you cut a slice.

Your rotary switch is mounted in a second disc of wood with a hole in the center, as in Figure L. Switch threads are not usually long enough for material 1/4" thick, so you'll probably have to glue it into place.

Figure M shows the pieces I've described so far, with a battery secured to the center piece using double-sided velcro tape. On the left, the electronic components are mounted on perforated board glued to a semicircle of plywood.

Figure N is a cross-section showing how the parts of the Decider are assembled, and Figure O shows them seen from above, before I added the top panel.

ELECTRONICS

I suggest you start by breadboarding the circuit, using the layout in Figure B, and read about the pinouts of the chips in datasheets online, to help with debugging the circuit if it doesn't work initially. Note that the actual part number of the 74HC4017 may be something like CD74HC4017E, but so long as the sequence 74HC4017 is in there, the chip will do what you want. (You'll find more information about this chip in my book *Make: More Electronics*.)

To make a permanent version of the circuit, small enough to fit the PVC tube, I soldered the parts with point-to-point wiring. The top side and underside of the board are shown in Figure P. Note that the underside is shown flipped left-to-right. You may want to use a hacksaw to cut the board, as its glass-fiber content can blunt saws made for wood.

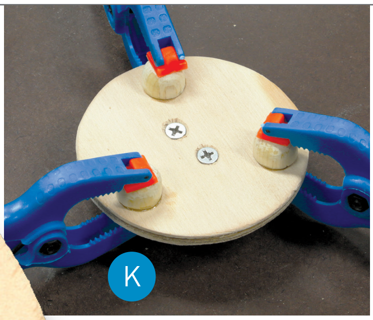
I put a couple of headers on the board so that I could plug pieces of ribbon cable into it, but you can solder the wires connecting the LEDs, the pushbuttons, and the rotary switch directly into the circuit if you prefer.

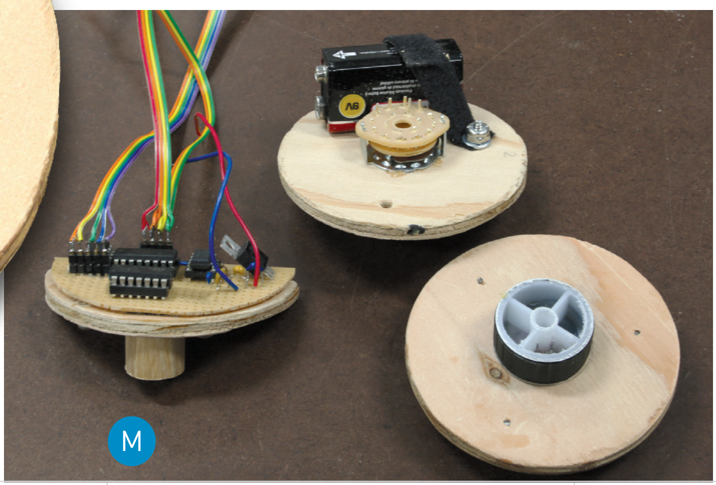
Because the LEDs light up one-at-a-time, a single series resistor is sufficient to protect all 3 of them. This is the 330Ω resistor shown in Figures B and P. However, if you use 3 colors of LEDs, each requires a different forward current for optimal performance. To get around this, use LEDs with their own series resistors built in, such as the Chicago Miniature series rated for 5VDC. Now you can ground the LEDs directly, eliminating the 330Ω series resistor, and all 3 will be equal in brightness.

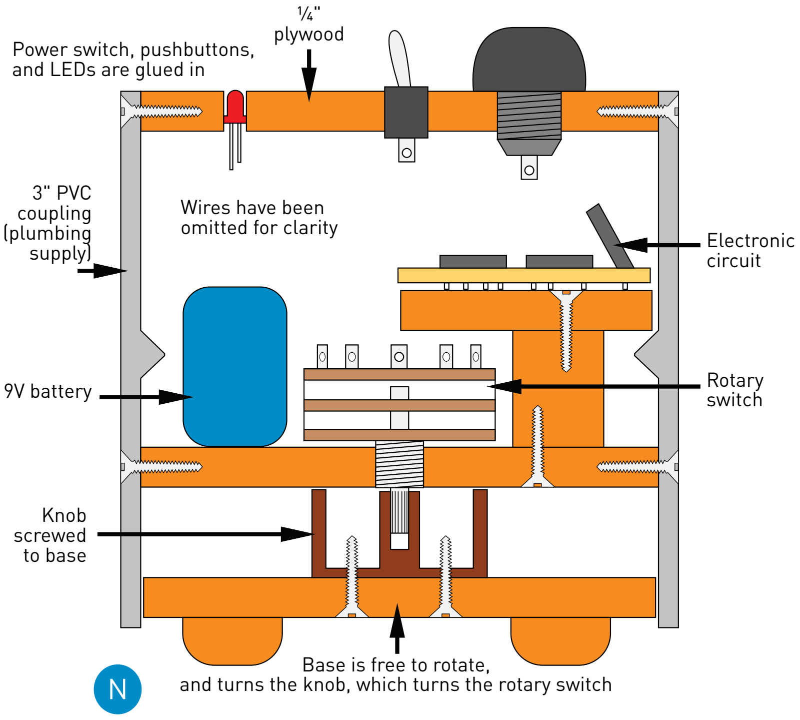
USING AND ABUSING IT

When you have assembled your Dishonest Decider, the base should turn freely, so long as the switch and the knob are properly centered. With some practice, you can

Charles Platt



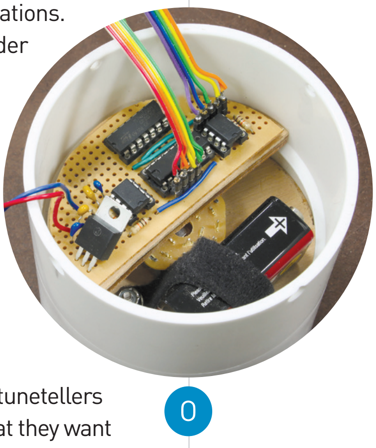




manipulate it without anyone noticing.

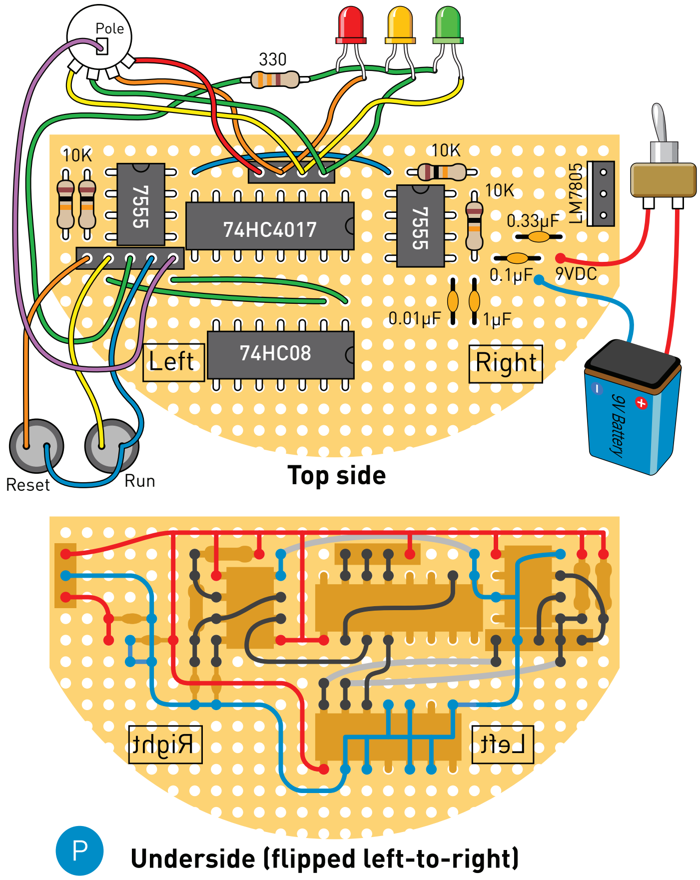
There's an obvious temptation, here: to use the Decider for rigged gambling purposes. This bothers me, not just because fraud is illegal, but because I don't want to be responsible if you get beaten up when someone figures out that you've been cheating.

I prefer harmless applications. For instance, use the Decider as a magic trick. In your hands, it always lights the green LED, but when you give it to someone else (after quietly turning the base), it always lights red. Take it back, and it goes back to green. Very mysterious!



You can also use it for fortunetelling. After all, fortunetellers traditionally tell people what they want to hear. The same is true in business, where an insecure manager may want independent validation to reassure him that he's doing the right thing.

In situations like these, the Dishonest Decider can keep everyone happy — so long as no one guesses its clever little secret.

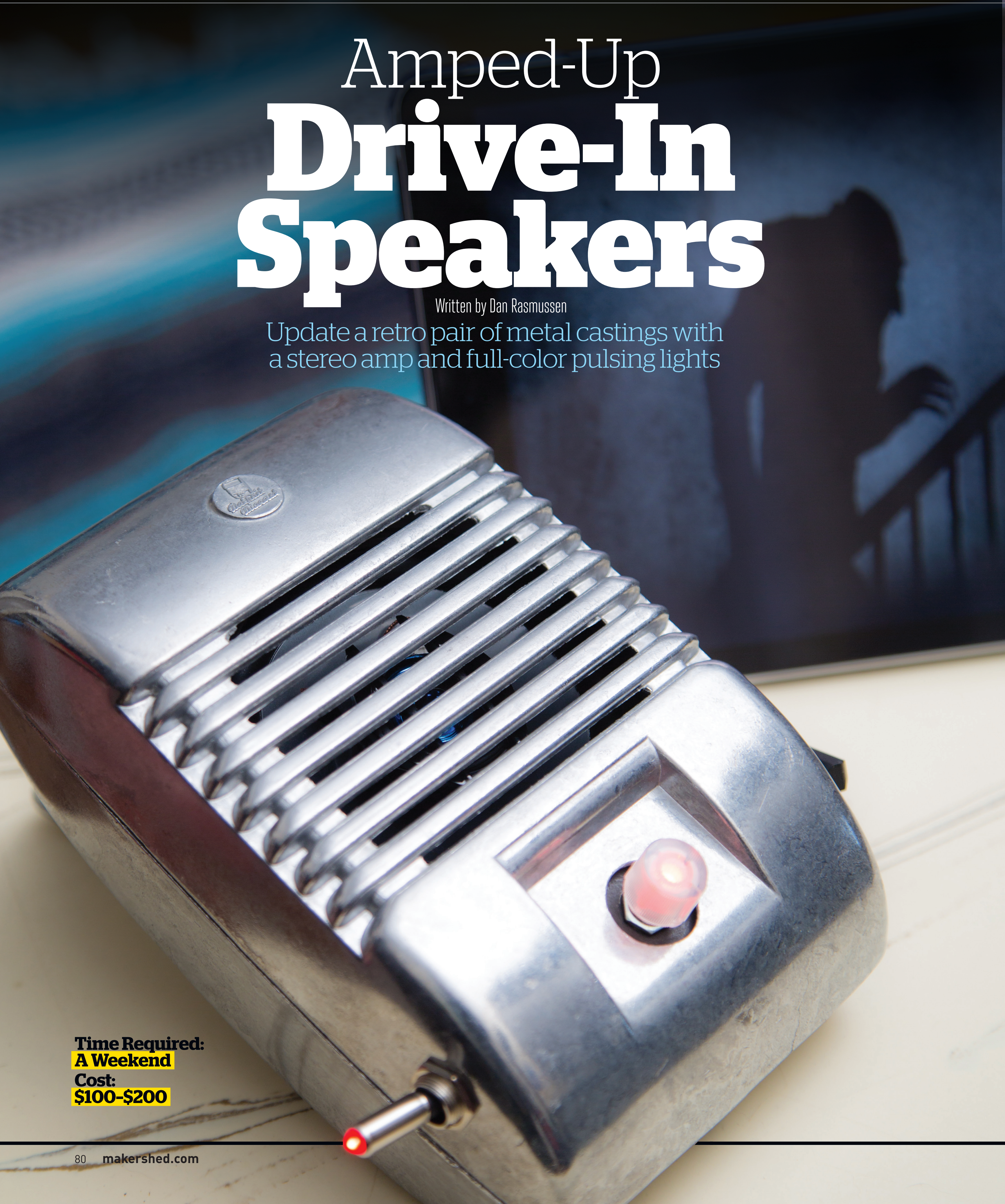


What tricks could you play using the Dishonest Decider? Share your build, and your dastardly deeds, at makezine.com/go/dishonest-decider.

Amped-Up Drive-In Speakers

Written by Dan Rasmussen

Update a retro pair of metal castings with a stereo amp and full-color pulsing lights



Time Required: A Weekend

Cost: $100–$200

DAN RASMUSSEN is an avid collector, fixer, and hacker of vintage technology. He works as a software engineer for a medical devices company and lives in Groton, Massachusetts, with his wife, two kids, and a dusty dog that bites just to kill the time.

WHEN I WAS A KID WE WENT TO THE DRIVE-IN THEATER TO SEE MOVIES LIKE *THE LEGEND OF BOGGY CREEK* AND *EVEL KNIEVEL*. I loved those movies but I remember just as vividly those retro-cool metal speakers that entered into our car for these campy films. A few years ago I found one of these speakers at a flea market and it came home with me.

After it sat in my garage for years, I finally decided it needed to be what it was made to be: a speaker. But this time it gets some pretty nice power and technology, as a 20-watt amplified stereo speaker set with an RGB lighted dial that first dials in your volume by color and then, with a click, pulses to the beat of the music. It's all done with 4 modern off-the-shelf circuit boards inside — but to keep the retro look, there's an old-school toggle switch for power.

1. BUILD THE ADAFRUIT AMPLIFIER

Follow the manufacturer's instructions for Digital Input, except don't install all the header pins; install only those for SDA, SCL, Vi2c, SHDN, Mute, GND, and VDD (Figure A).

2. PREPARE THE ARDUINO PRO MINI

Solder the headers on (Figure B), then download the project code file *DriveIn.ino* from the project page at makezine.com/go/amped-drive-in-speakers, and upload it to the Arduino.

3. SOLDER RGB ENCODER TO BREAKOUT

Insert the encoder into the side of the board labeled RGB, then solder the headers on the opposite side, labeled RG (Figure C).

4. SOLDER HEADERS TO MIC BOARD

Solder the 3 header pins on (Figure D).

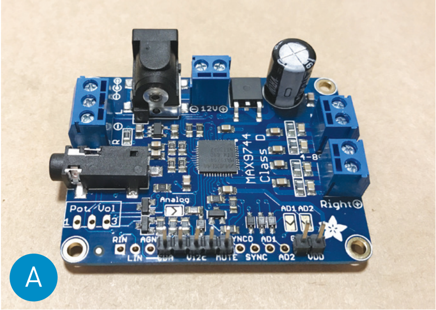
5. WIRE THE POWER CONNECTIONS

Wire the toggle switch and barrel jack following Figure E. Note that the switch has been wired to illuminate only when power is switched on. It can also be wired to always illuminate (see Adafruit's instructions).

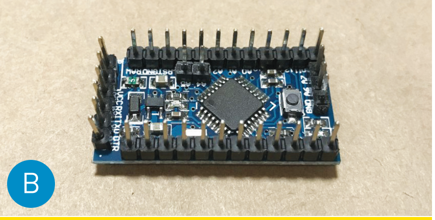
Connect the loose power and ground wires to the amplifier board's 12V + and – screw terminals. Later we'll use the amp's VDD output pin to power the Arduino.

6. MOUNT THE BOARDS FOR TESTING

You can use the Scotch fasteners to attach

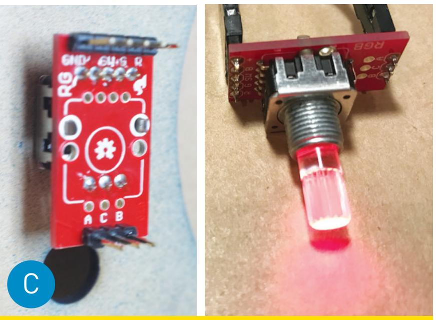


A



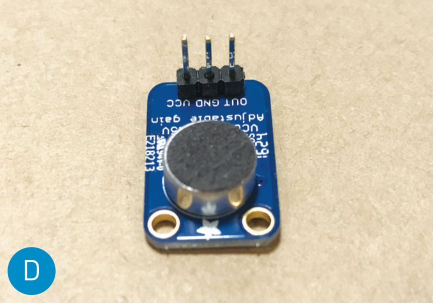
B

TIP: Find or draw a diagram of the pin markings on the Arduino, because they're difficult to see after the headers are installed.

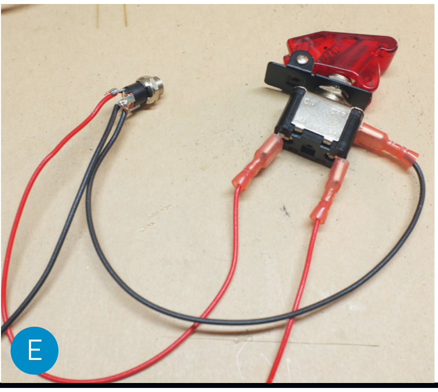


C

NOTE: It's easy to mistakenly use the wrong side of the encoder breakout board. Be sure to assemble as shown.



D



E

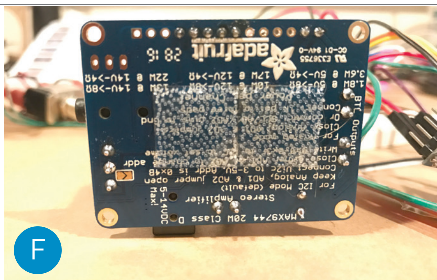
Hep Svadja, Dan Rasmussen

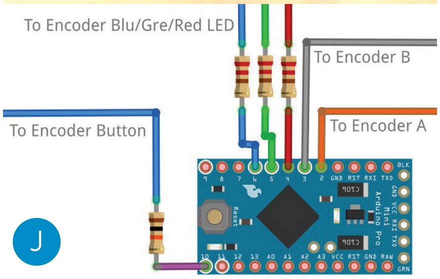
MATERIALS

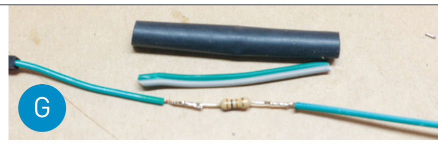
- **Rotary encoder, illuminated (RGB)** SparkFun #10982 sparkfun.com
- **Breakout board** for rotary encoder SparkFun #11722
- **Clear plastic knob** SparkFun #10597
- **Arduino Pro Mini 328 microcontroller, 5V/16MHz**
- **Stereo audio amplifier, 20W class D** Maxim #MAX9744, Adafruit #1752 adafruit.com
- **Full-range speakers, 20W 4Ω (2)** Adafruit #1732
- **Toggle switch with cover, illuminated** Adafruit #3218
- **Electret microphone amplifier with adjustable gain** Maxim MAX4466, Adafruit #1063
- **Jumper wires, 12" female to female (25)** such as Adafruit #793
- **RCA drive-in movie speaker castings (2)** Find originals on eBay, or buy reproductions from Rich at Detroit Diecast (detroitdiecast.com).
- **Mounting panels, ⅛" fiberboard** You can cut them from the template at makezine.com/go/upgrade-drive-in-speakers, or buy them from my shop, Retro-Tronics (makezine.com/go/drive-in-panels).
- **DC barrel jack, panel mount** SparkFun #10785
- **Scotch all-weather fastener strips** Amazon #B00347A8EO
- **DC power supply, 12V**
- **Audio cables, stereo, 3.5mm plug/plug, 6' length (2)** Adafruit #876
- **Cable tie mounts, self-adhesive (4)** Amazon #B00SN1BS8G
- **Resistors: 10kΩ (1), 150Ω ¼W (3)**
- **Heat-shrink tubing**
- **Speaker wire or other 2-conductor cable**
- **Rubber grommets (2) (optional)** to fit your speaker wire and audio input cable
- **Machine screws, 8-32 (optional)** for reproduction speaker housings
- **Paint, your choice (optional)** for speaker housings. For vintage housings, you could also try blasting them, or embrace the patina.
- **Paint, flat black (optional)** for panels
- **Tape**

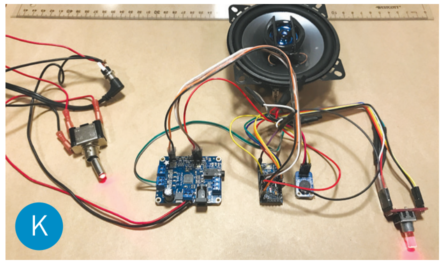
TOOLS

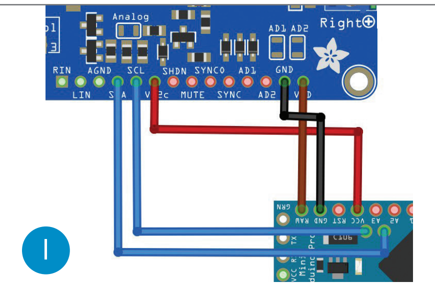
- **Soldering iron and solder**
- **Computer with Arduino IDE software** free download from arduino.cc/downloads
- **USB cable, FTDI serial TTL-232** for programming the Arduino Pro Mini
- **Drill and twist bits**
- **Wire cutters/strippers**
- **Gloves**
- **Voltmeter**
- **Screwdrivers**
- **Tap set, SAE 8-32 (optional)**
- **Tapping oil or WD-40 (optional)**
- **Dremel with mini cut-off wheel (optional)**

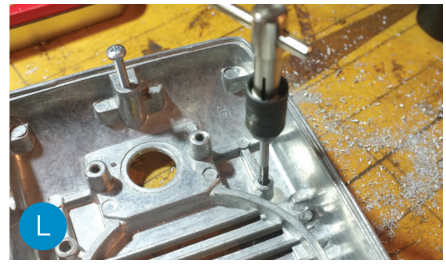












your boards to a work area for wiring and testing. Apply a section of fastener to the underside of each board (Figure F), then apply another fastener to a test mounting board (some stiff cardboard is fine).

You'll use the same fasteners to mount these inside the speaker housing.

7. PREPARE THE ENCODER WIRES

The RGB jumper wires each require an in-line resistor, 150-ohm ¼-watt. Assemble as shown in Figure G, then seal the joint with heat-shrink tubing. I like to add an extra piece of wire (disconnected) inside the heat-shrink to stiffen the joint.

Additionally, a pull-down resistor is required for the encoder's integrated switch. This one is a 10K. Assemble as shown in Figure H, then seal the joint with heat-shrink tubing.

THE HARDWARE

ARDUINO PRO MINI: As with most of my hardware projects, I prototyped this one with the standard Arduino Uno board, but that won't fit into the drive-in speaker case. So I chose the Arduino Pro Mini — inexpensive, very small, and reliable.

ADAFRUIT STEREO 20W AUDIO AMPLIFIER: One thing I really needed for this project was the ability to control the volume digitally. I settled on Adafruit's beautiful little MAX9744 amp board — it's got everything I need in a small package and it sounds awesome.

ADAFRUIT ELECTRET MICROPHONE AMPLIFIER: This optional finishing touch provides the modulation of the color of the LED, to the beat of the music. (If you don't want this mode, you can simply omit the mic from the project and everything else will work just fine.)

SPARKFUN RGB ENCODER BREAKOUT BOARD: Makes the addition of the RGB encoder easy.

8. CONNECT THE ELECTRONICS

Follow the diagrams in Figures I and J to connect the amp, Arduino, encoder board, and RGB LED. Be as neat as possible; it will make final assembly much easier.

Connect your speakers to the amplifier's Left and Right screw terminals. Connect your (optional) microphone board's GND, 5V, and Out pins to the Arduino's GND, 5V, and Analog pin 2, respectively.

After the wires are in place, use tape to manage the plug ends and the wires. It's important to get the wires from the encoder board nice and flat, so they're easy to reconnect. I used packing tape to group them into a little ribbon cable.

9. CHECK YOUR WORK

Verify all your connections, then plug the DC power supply into the barrel jack. The switch should light up when you turn it on. You should also see the LED encoder knob light up (Figure K) and cycle through red, green, and blue before settling on its default volume color level — a good indication that all is well.

Plug an audio input source into the amplifier board using one of the audio patch cords. Be sure your input source volume is turned up to a moderate/high volume. Now you should be able to control the speaker output volume by using the encoder dial.

Try turning the encoder's shaft — you should see it change color. Click the encoder, and you should see the light modulate with the music (if you're using the mic). Click again and it should to return to your solid volume color.

10. PREPARE THE SPEAKER HOUSINGS

If you're using vintage housings, disassemble them and remove all components. You only need to retain the housing and screws.

If you're using reproduction housings, you might want to tap the speaker mounting holes to accept machine screws (Figure L). (You can use the self-tapping screws that the housings came with but these may eventually strip out.) I got a cheap tap set from Harbor Freight and used the metric 8-32 tap and M8 screws. Go slow, and use a bit of tapping oil or WD40 for lubrication. Remove and clean the tap often as you go.

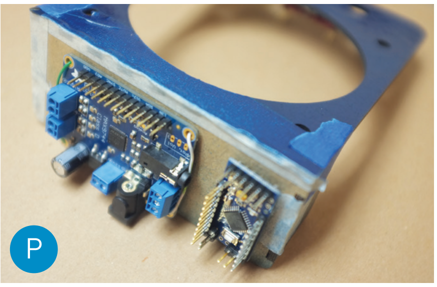
TIP: Some vintage speakers were assembled with security screws — these may have to be drilled out, or they may be big enough that you can cut a slot in the top for a screwdriver, using a Dremel and mini cut-off wheel.

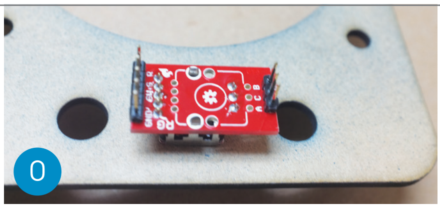
11. DRILL THE LEFT HOUSING

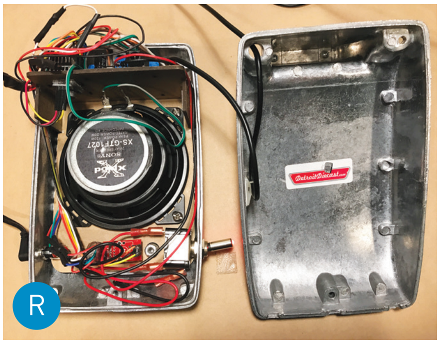
The left speaker housing needs 3 holes drilled in the front half of the casting (Figure M), sized to fit your hardware and speaker wire: at bottom left for the on-off power switch, bottom right for the power jack, and top right for the wire to the second speaker. Placement doesn't have to be perfect.

TIP: Because these aluminum castings can crack or shatter, don't use a punch to mark the location of the holes. Instead, mark them with a Sharpie, carefully make a dimple using a very small drill bit, then finish each hole with the final bit. These castings drill easily using a nice sharp bit and moderate pressure.

If you're going to paint your speakers, now is the time. Apply 2 thin coats of primer and







then 2 or 3 thin coats of paint (Figure N).

Optionally, add a grommet to the speaker-wire hole and the original rear hole (for your audio input cord) for a more finished look.

12. MAKE THE MOUNTING PANELS

To fit the circuit boards neatly into the left housing, you'll use mounting panels made from ⅛" (0.125" or 3mm) fiberboard. Download the template from makezine.com/go/upgrade-drive-in-speakers to cut your own, or buy them from my shop at makezine.com/go/drive-in-panels.

For the left speaker, tape or glue the small horizontal shelf to the large vertical panel, using the 2 corner braces. For the right speaker, you only need a vertical panel.

I recommend painting the front flat black to prevent it showing through the grill.

13. MOUNT THE ELECTRONICS

On the vertical panel, mount the encoder in the center hole at the bottom (Figure O) using its included nut. (It's easiest to disconnect the encoder and mic boards before mounting.) This panel also serves as a spacer, since our speakers aren't exactly the same size as the vintage ones.

On the horizontal shelf, use the Scotch fasteners to mount the Arduino and amplifier board on top (Figure P), and the mic board underneath. Place the amp board carefully to avoid interference with the top of the case.

Remove the 3 circuit boards temporarily and slide the mounting panels into the case. Then mount the speaker, DC power jack, and power toggle switch.

14. CONNECT THE RIGHT SPEAKER

Mount the right speaker in its housing, using a second vertical panel as a spacer.

In the left housing, attach your speaker wire to the Right output terminals of the amplifier board, and run it through a self-adhesive cable tie mount to provide strain relief (Figure Q). Then thread it out through its grommetted hole, and solder it to the 2 hookup wires provided with the speaker.

15. ADD THE AUDIO INPUT CORD

Cut one end off an audio patch cable and thread it through the original audio cord hole in the back half of the casting. Again, use a cable tie mount for strain relief. Then connect its 3 wires to the R/–/L audio input screw terminals on the amplifier board, using your voltmeter to determine which wire goes to the tip/ring/sleeve of the cable (test for near-zero resistance). Attach the tip to L, the ring (next section behind the tip) to R, and the sleeve to ground (–).

Press your circuit boards back into place (Figure R) and double-check connections. Close up the cases, push the clear knob onto the encoder shaft, and enjoy!

PASSION PIT PLAYLIST

Hang your drive-in speakers somewhere fun, then connect your audio source. This little 20W amp provides a lot of sound!

The encoder dial will change colors as you turn the volume up and down. Now click it to make the light pulse to the beat; the little microphone board listens to the music, then the Arduino filters the analog signal to detect the beat (thanks to some open source software) and modulate the light.

GOING FURTHER

I considered Bluetooth wireless audio input, but found that it was a relatively expensive and complex addition to the project. You could try plugging a car Bluetooth adapter (e.g. Amazon #B00LVFPXNC) into the amp board, but I suspect the metal housing might interfere. I'd love to see your solution!

Either way, these amped-up drive-in speakers are a super cool way to groove to some retro (or not so retro) tunes.

See more photos, videos, and share your build at makezine.com/go/amped-drive-in-speakers.

Dan Rasmussen, Hep Svadja

1+2+3 Rainbow Flowers

Written by Lisa Martin

Hep Svadja

IF YOU WANT TO GET CREATIVE WITH FLOWERS, there's another option besides simple arranging: tie-dye! You can dye white blooms to create multi-colored blossoms.

1. TRIM THE STEMS

Cut the stems of the flowers so they're between 12"–18" tall (the shorter the stem, the faster the color change). Remove any large leaves. Using a sharp knife, cut a 6" slit that bisects the very bottom of the stem. You should now have 2 sections of stem at the bottom of your flower. If you're using more than two colors of dye, cut one or both of these stem sections in half. Keep these cut edges moist since exposure to oxygen can make the flowers wilt at a faster rate.

2. ADD DYE

Set up a container for each color you plan to use and fill with enough water to cover the slit you cut in the stems. Use roughly 20-30 drops of dye per cup. If your bouquet came with a little packet of flower food, you can divide it equally among the cups. Place each separate stem end into a cup of colored water and prop up the flowers so they don't fall over.

3. WAIT

You should begin to see the first hints of color in the petals after a couple of hours, but wait a full 24 hours to see an even more vibrant color change. Don't wait too long though — these flowers won't last forever. Cut off the frayed bottom edges and gift your tie-dye bouquet to someone special.

Time Required: 8-24 Hours

Cost: $11-$20

MATERIALS

» **White flowers**
» **Food dye (2 or more colors)**
» **Water**

TOOLS

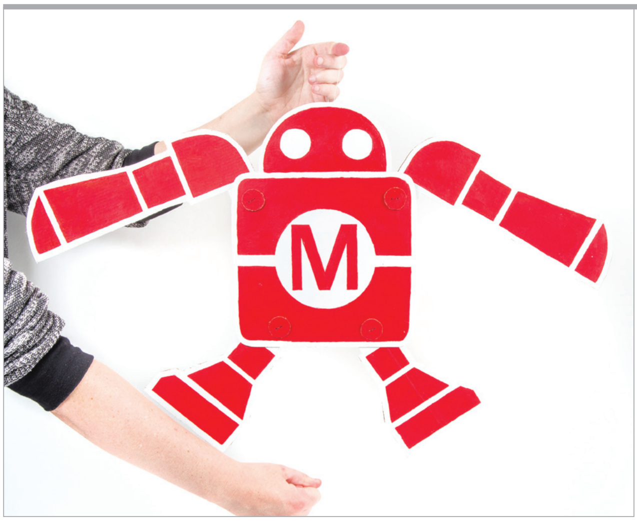
» **Knife or scissors**
» **Containers,** one for each dye color
» **Tape**

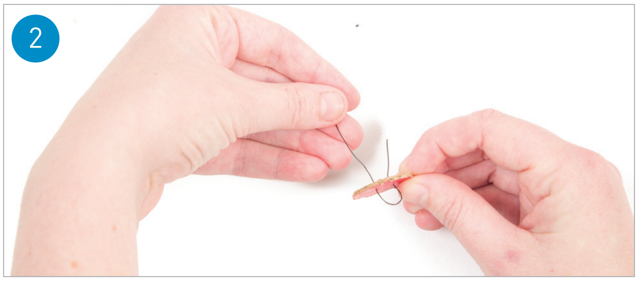
LISA MARTIN is the *Make:* editorial intern. She's no expert, but has grown, eaten, dyed, and killed many flowers in her time.

SCIENCE! As the flower releases moisture through a process known as transpiration, more water gets pulled up through tiny tubes within the stems known as xylem. Water has a tendency to "stick" to water, so as a water molecule leaves the plant it drags the next water molecule up to take its place.

1+2+3 Jumping Jack Makey

Written by Lisa Martin





Hep Svadja

YANK THE STRING ON THIS SIMPLE PUPPET AND THE ARMS AND LEGS GO UP. It's a straightforward design that can be made at home with cheap and easy-to-find supplies.

1. CUT OUT THE PIECES

There are 5 parts: two legs, two arms, and the head/torso. Create your own character design or trace a pattern onto your cardboard, then paint. Use an awl to poke holes through the pieces where you want them to connect.

2. INSERT RIVETS

Rivets let the arms and legs move freely. You can use brads or box rivets for this purpose, but you don't need to buy special hardware. I decided to jury-rig some wire-and-cardboard rivets. Cut out four small cardboard circles. Use your awl to poke two holes into each circle (like buttons), then carefully bend short segments of wire to go through the holes and wrap them together on the other side. Use the rivets to attach the legs and arms behind the torso.

3. ADD STRINGS

Place the arms and legs as you would like them to hang in their starting positions. Connect the two arms together with one string, making sure there is no slack and that the ends aren't too close to the rivet. Tack down each end with glue. Use a second piece of string to do the same to the legs. A third string should be tied to both the arm string and the leg string, with a long tail dangling below your jumping jack. Now when you pull the string the arms and legs will pop out. Attach a loop of string to the very top of the jumping jack to hang or hold it.

Share your jumping jack toy builds at makezine.com/go/jumping-jack-toy.

Time Required: 60 Minutes
Cost: $0–$20

MATERIALS

- **Cardboard**
- **String or fishing line**
- **Wire, brads, or box rivets** such as Mr. McGroovy's Half & Half kit mrmcgroovys.com
- **Paint**

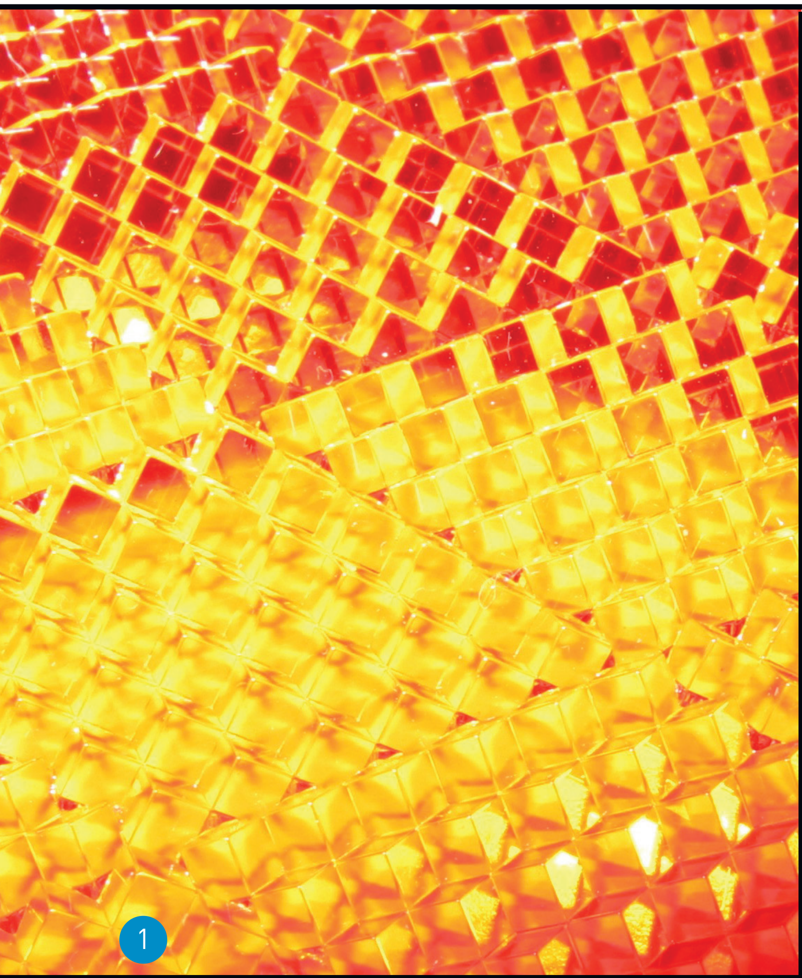
TOOLS

- **Scissors or craft knife**
- **Awl**
- **Glue**

LISA MARTIN
Is the *Make:* editorial intern. She spends more time painting on cardboard than canvas, but still likes to flex those artistic muscles.

CUSTOMIZE!

Design your own jumping jack character or use enlarged photos of your friends to piece together. Traditionally, these dolls have been made from both paper and wood, but the same design will work with other materials. I did mine in cardboard — what fun materials can you find to make yours from?


1


2


3

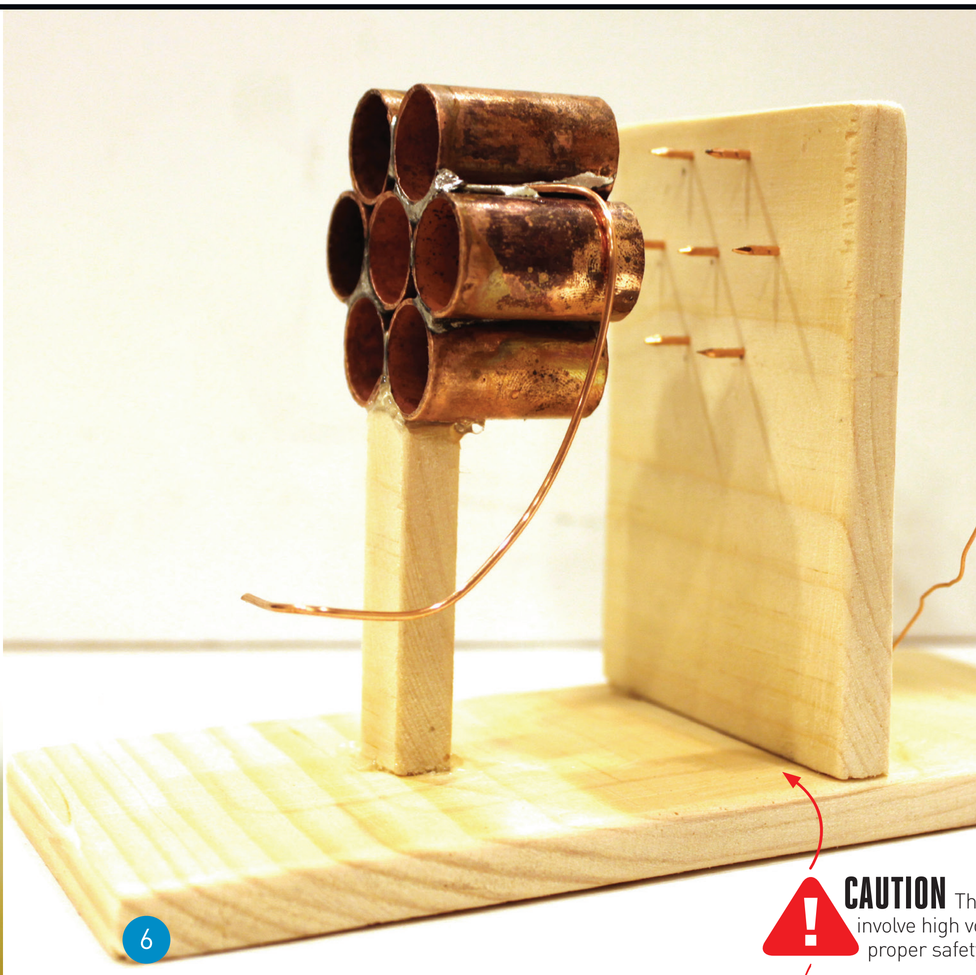

4


5

Experimentation Inspiration

Written by Caleb Kraft

Fun projects to spark your ideas for this year's science fair



CALEB KRAFT is Senior Editor at *Make:*. While he is always up for a good experiment, proper documentation eludes him.

CAUTION These two projects involve high voltage and require proper safety precautions

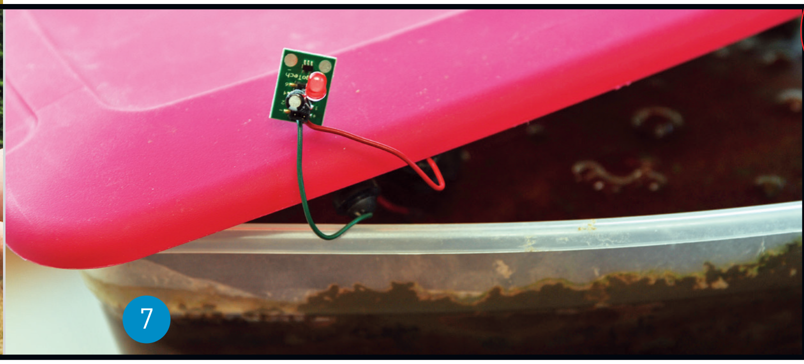
KNOW THE SCIENTIFIC METHOD

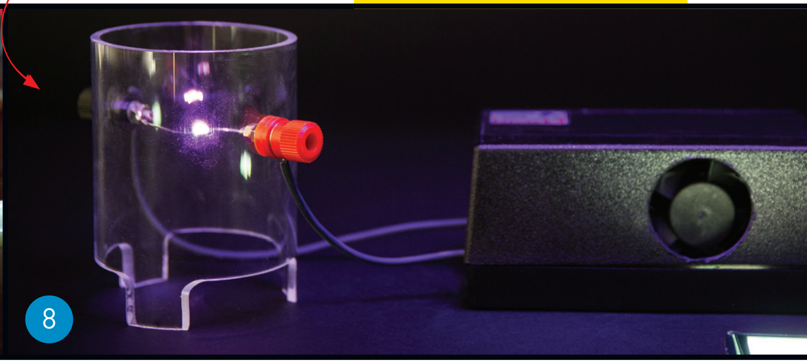
The key to any good experiment is to follow the tried-and-true scientific method. This set of steps, which dates to ancient Greece, has various elements, but four key stages to incorporate at the very least:

» Establish your question
» Form a hypothesis
» Predict an outcome
» Test the hypothesis

Remember to set up a solid control group when you're testing your hypothesis — this is crucial to the analysis of your experiment to show your results.

Forrest Mims, Sam Murphy, Make Labs, Andrew Sullivan, Gregory Hayes, Alexander Reifsnyder, Gregory Hayes, Hep Svadja





SCIENCE FAIR SEASON IS QUICKLY APPROACHING! The very first step to any good fair experience is to pick an awesome experiment to do, and then apply the scientific method to determine the results.

Here are a few how-to projects to help inspire a fun trial for you to do at your fair. Use these to develop a hypothesis, put it to the test, record the results, then share with the rest of our readers at community.makezine.com!

1 REFLECTING LIGHT
makezine.com/go/retroreflectors
Build a physical display that demonstrates the property of retroreflection at work.

2 FULLY ENCLOSED BIOSPHERE
makezine.com/projects/tabletop-biosphere
Create a micro ecosystem in a jar.

3 GOOEY OOBLECK
makezine.com/go/oobleck
This slimy goop isn't just fascinating fun, it's also a perfect example of a non-newtonian fluid.

4 BICYCLE GENERATOR
makezine.com/projects/generator-bicycle
Create electricity by pedaling on a stationary bike. This impressive project can be built for under $100.

5 MAGDEBURG HEMISPHERES
makezine.com/projects/the-magdeburg-hemispheres
Show how one 17th-century scientist displayed the incredible power of vacuum with this clever experiment.

6 IONIC SPACESHIP THRUSTER
makezine.com/projects/ionic-thruster
Create a functional engine with no moving parts, like what NASA uses in space.

7 BACTERIA BATTERY
makezine.com/projects/bacteria-battery
Power an LED with nothing more than a fancy bucket of mud.

8 PLASMA ARC SPEAKER
makezine.com/projects/plasma-arc-speaker
Watch as a tiny bolt of lightning zaps music into the air.

TOOLBOX

GADGETS AND GEAR FOR MAKERS

Tell us about your faves: *editor@makezine.com*

DEWALT CORDLESS ANGLE GRINDER **$120** dewalt.com

When I'm out in the field, or I just can't find a free outlet around the house, it's nice to have the option of a cordless grinder. My DeWalt DCG412B delivers a ton of power without the hassle of dragging around a cord or searching for power on the go. The absence of cord also enables me to get into small, awkward spaces and burrow into cramped quarters, like inside a car body.

The battery life is good considering the torque, but if you are doing a lot of serious cuts, be sure to have several charged spares on hand. At 8,000 RPMs, the motor is beefy whether you are grinding stone or cutting metal. It's easy to hold and manage even for a smaller-sized human like myself, and the "Quick-Change" feature makes swapping wheels fast and efficient.

I've owned this model for over a year, and it's handled jobs every weekend from cutting tile, buffing and polishing metal, to making thousands of metal cuts in the blacksmith shop. It's definitely a great addition to a field kit. *—Hep Svadja*

Hep Svadja

TECHNICIAN'S POCKET SCREWDRIVER

$20 countycomm.com

When repairing and tuning electronics on the go, it helps to have a little screwdriver set. The Maratac Technician's Screwdriver is my favorite for this kind of assortment.

This anodized aluminum, pen-style driver has a reversible bit on each end under screw-off caps. Flip the bits over to chose between a 1/16" or 1/8" slotted and a P00 or P0 Philips. Unlike many inexpensive pocket driver sets, these won't bend or deform easily, as they're made from heat-treated tool steel.

One of my favorite features is the included neodymium magnet on the end of one cap for storing small screws as you work. This driver feels great in your hand and is a joy to use. *—John Edgar Park*

ALPHATIG 200X TIG/STICK WELDER

$800 ahpwelds.com

As far as affordable, entry-level TIG options go, the Alpha 200X TIG and stick welder is an excellent choice. It operates on both 110V and 220V, using an IGBT and PWM-based inverter. As both an AC and DC unit, it can weld up to 1/4" aluminum and 3/8" mild steel. The unit ships complete with everything except consumables such as electrodes, rods, and argon. The 2016 model includes an improved torch, foot pedal, and an actual ball and tube argon flow meter/regulator. The 2016 model also expands the pulse modulation range from 0.5Hz–5Hz to 0.5Hz–200Hz.

With 9 knobs and 3 switches, the 200X can be a little daunting, and the very basic manual is not a tutorial. As a self-taught novice, I was able to learn a lot from some excellent YouTube videos, and AHP support was extremely helpful when I had a problem with my flow meter.

The AHP 200X has dramatically improved my stick welding as well. Upgrading from a lunchbox stick welder to one that has real power means I'm struggling less than I used to. This, and the dual voltage makes the 200X tremendously versatile. This is an impressive entry into a welding process that can do amazing, beautiful welds on aluminum, stainless and mild steel. *—Tim Deagan*

SHAPECRETE

$30 shapecrete.com

Everyone at *Make:* was tempted by the bucket of ShapeCrete in our office, but I was the lucky one who lugged it home. It's easy to work with in small portions and isn't very messy. (With just a balcony for outdoor space, there's no way I'd bring home normal concrete.)

While I wouldn't recommend it as a perfectly clay-like substitute, it's a fine choice for molding. I didn't plan to use it that way, but when my first batch turned out wetter than I wanted (pro tip: add water incrementally), I improvised and poured it into a clean salad bowl. Not only did it release from the plastic easily, it also perfectly retained all the small recycling info imprinted at the bottom of the bowl.

Even at its thickest consistency, it isn't quite stiff enough to hold its own weight, so if you have experience working with clay, don't expect ShapeCrete to behave the same way. To get the most out of it you'll want a mold, either as a container or as a shape to form the mix onto. *—Lisa Martin*

SNAP CIRCUITS MOTION

$85 snapcircuits.net

Snap Circuits is a fun kit that teaches electronics with solderless, snap-together electronic components. Each Snap Circuits block contains a component (such as a capacitor) with the schematic symbol and part number printed on it for easy identification. In order to assemble the circuits, the full-color manual features easy-to-follow instructions for snapping components onto the 10×7 plastic base. When the project is completed, it will look like an electronic schematic.

The Snap Circuits Motion set includes electronic components to build dozens of circuits as well as motors, gears, and pulleys for gear ratio experiments. The set also includes a motion detector, air fountain, color-changing LED fan, mini-car, and robotic crawler. *—Steve Schuler*

Petr Krejci Photography

KANO COMPLETE

$300 kano.me

The Kano Complete offers both a snap-together, Raspberry Pi 3-powered computer kit and a build-your-own screen kit. Assembling and programming is easy enough for makers as young as 6 to learn how computers work and jump into the basics of computer programming.

Once the Kano is powered up, users can learn how to program right away with a customizable version of Minecraft using Kano Blocks, a graphical drag and drop programming language where the code is assembled like Lego bricks. Once the blocks are put together, they can be converted to Python or JavaScript, or simply run as-is to see the changes made to the Minecraft world.

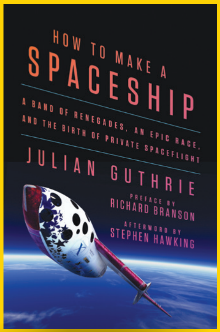
The Raspberry Pi's GPIO pins can also be programmed using Python or MIT's Scratch to control external electronic circuits or Pi HATs. *—SS*

BOOKS

HOW TO MAKE A SPACESHIP

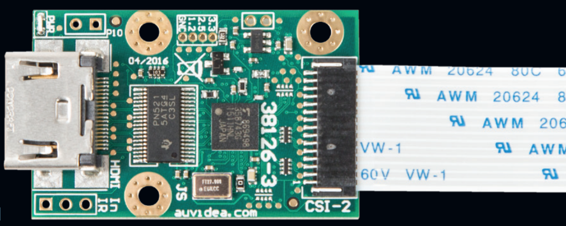
BY JULIAN GUTHRIE

$28 penguinrandomhouse.com



The closest I'll get to being an astronaut is costumes and Photoshop. Luckily for me, I am living in the age of private space initiatives and commercial ventures to bring space to consumers, where each day brings new breakthroughs for private spaceflight. I have long studied our government space program, but, as someone who's had a Google alert for "space travel" for 10 years, I found myself pretty uninformed about a lot of private space history. *How to Make a Spaceship* chronicles the journey to commercial space through the efforts of Peter Diamandis, Mojave Aerospace Ventures, the SpaceShipOne, and other teams who competed for the first XPrize. From young Diamandis growing up, to the tense competition for the Ansari XPrize, to the final flights where Mike Melvill and Brian Binnie each completed successful flights of the SpaceShipOne, this book never lets up escape velocity.

The inspiring anecdotes will keep you motivated to finish your own dream projects. Filled with amusing and interesting rocketry trivia, this book is a great read for any maker or space fan, especially for rocketry hobbyists with delta-v dreams above the Kármán line. *—HS*



B101

$79 auvidea.eu

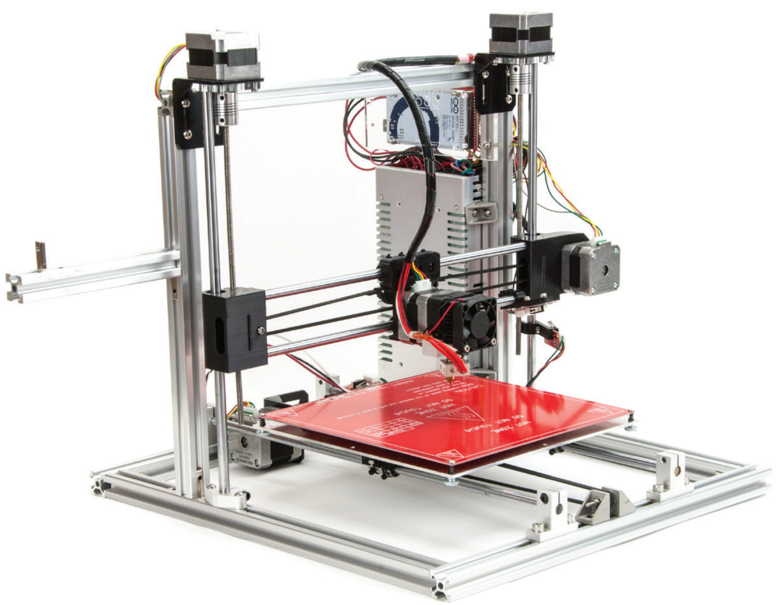
The B101 by Auvidea is an add-on for the Raspberry Pi that allows it to capture video from an HDMI source. Instead of using an array of GPIO pins, the B101 instead makes use of the CSI-2 bridge — the same connection that the Pi uses to communicate with the Raspberry Pi camera. Its resolution currently tops out at 1080p/25, but that's just a software limitation. Capturing footage from HDMI sources opens up a whole passel of projects — turning the Raspberry Pi into a recording device, a video-over-IP bridge, or a rudimentary livestreaming system.

—Tyler Winegarner

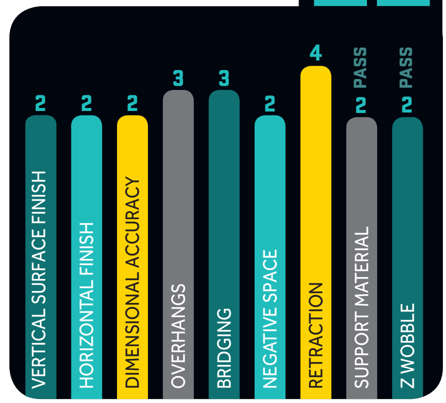
2020 PRUSA i3

With the right upgrades, this inexpensive machine can hold its own

Written by Matt Stultz



MACHINE RATING 22



folgertech.com

- **MANUFACTURER** Folger Tech
- **PRICE AS TESTED** $270
- **BUILD VOLUME** 200×200×150mm
- **BED STYLE** Heated bed (no surface supplied)
- **FILAMENT SIZE** 1.75 mm
- **OPEN FILAMENT?** Yes
- **TEMPERATURE CONTROL?** Yes, tool head (230°C); bed (120°C)
- **PRINT UNTETHERED?** Yes (SD card)
- **ONBOARD CONTROLS?** Yes (LCD with control knob)
- **HOST/SLICER SOFTWARE** Repeteir with slic3r
- **OS** Windows, Mac, Linux
- **FIRMWARE** Open Marlin
- **OPEN SOFTWARE?** Yes, both software and firmware
- **OPEN HARDWARE?** Yes, no license given

PRO TIPS

If you have access to another printer, make some of the upgrade pieces (like a print-cooling fan) found on sites like Thingiverse before assembling this machine. As a very bare-boned kit, they'll prove useful if not necessary.

WHY TO BUY

This is a super affordable option that gets you 90% of the way to what you want. While I think a few upgrades are in order, you are not paying for a ton of parts you are just going to throw out anyway.

Matt Stultz

THE FOLGER TECH 2020 PRUSA I3 IS A SUB $300 PRINTER YOU CAN ORDER today, and not have to wait for all the fun that comes with a crowdfunded printer. As they say though, you get what you pay for. In this case, it's not quite a complete printer.

MINIMALISTIC KIT

The 2020 Prusa i3's name is a take on the classic Prusa i3. Sporting an aluminum frame, the majority of the parts are metal, with only a handful of printed parts specific to 3D printers. This helps create a rigid but still very modular and hackable platform for users.

To help keep the cost down — and I'm sure in some cases to make shipping easier — Folger Tech has omitted a few items from this kit. You will need to supply your own build surface to help cover the standard RepRap heated bed that it ships with. If you plan on printing with PLA, a print-cooling fan is desperately needed as well.

Thanks to its simplicity, assembly is fairly straightforward and quick. If you've never built anything using aluminum extrusion before, don't rush to tighten down all your screws until after the construction is complete. It will be easier to square up the machine once everything is in place.

MINOR TWEAKS, MAJOR RESULTS

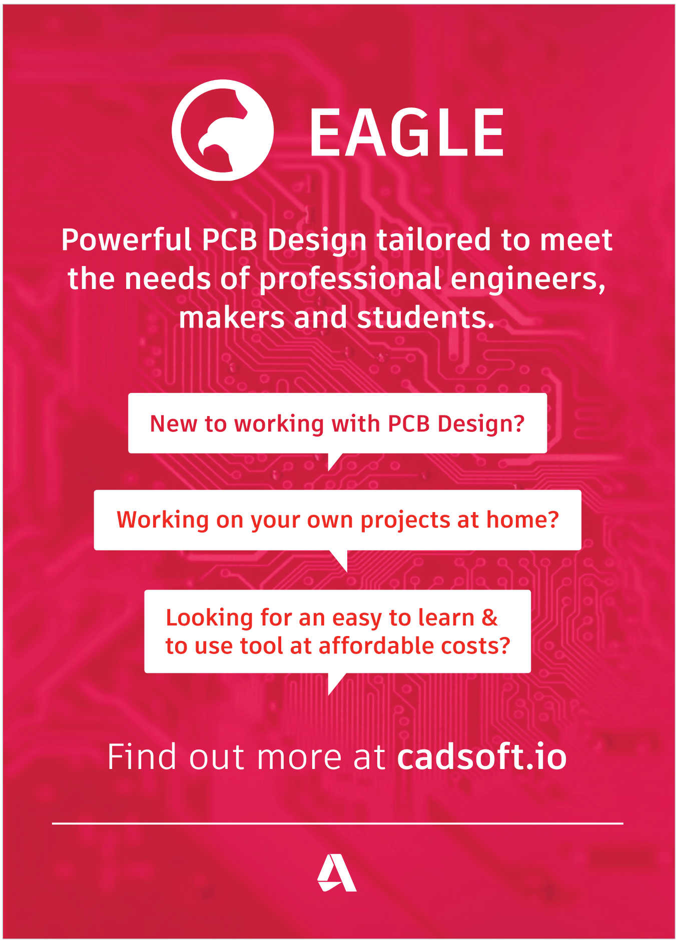
One big win for this machine is its inclusion of a heated bed, a rarity at this price point. This opens the door for this machine to be used with a much larger collection of filaments, not just PLA variants.

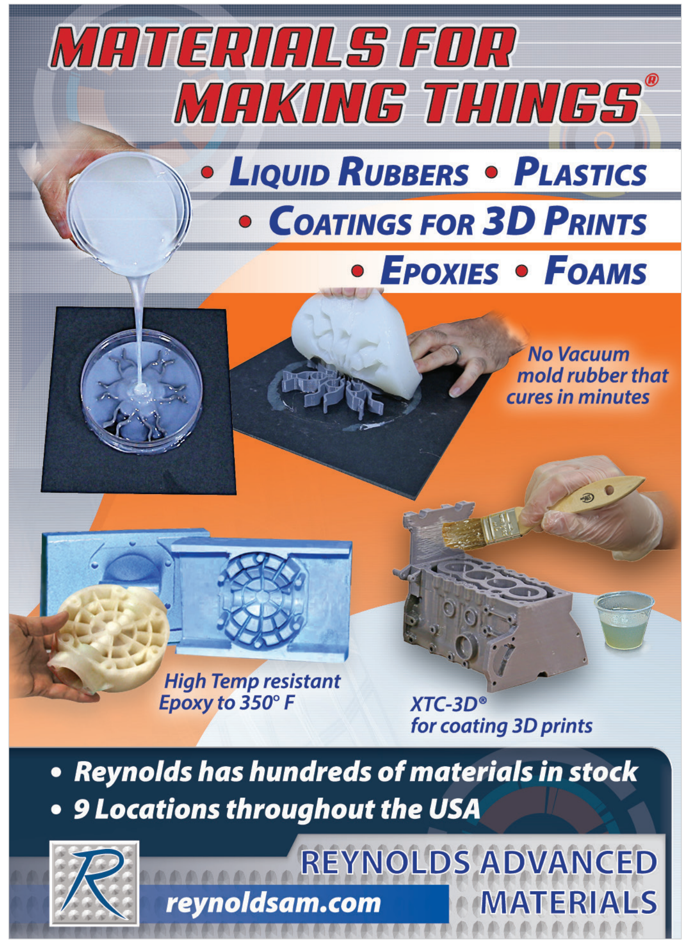
While the print quality was lacking in our tests, this issue is easily solved. A novice user is most likely to use the settings recommended by the manufacturer and might not know they could do better with different settings. The defaults for the 2020 Prusa i3 are its biggest detractor. Our scoring reflects the prints made with their settings, but I saw a huge improvement on the machine by simply opening Cura and using a default i3 profile with no other changes except for filament diameter and nozzle size.

LOW COST COMPETITOR

I think Folger Tech is onto something with their 2020 Prusa i3. While the price point will undoubtedly attract those interested in getting started with the hobby, it might not be the best machine for those jumping into the waters on their own. For an expert-lead group build though, this is a great consideration and with the right upgrades, the 2020 Prusa i3 could keep up with machines five times its cost.

MATT STULTZ is the 3D printing and digital fabrication lead for *Make:*. He is also the founder and organizer of 3DPPVD and Ocean State Maker Mill, where he spends his time tinkering in Rhode Island.


EAGLE
Powerful PCB Design tailored to meet the needs of professional engineers, makers and students.
New to working with PCB Design?
Working on your own projects at home?
Looking for an easy to learn & to use tool at affordable costs?
Find out more at cadsoft.io


MATERIALS FOR MAKING THINGS®
• LIQUID RUBBERS • PLASTICS
• COATINGS FOR 3D PRINTS
• EPOXIES • FOAMS
No Vacuum mold rubber that cures in minutes
High Temp resistant Epoxy to 350° F
XTC-3D® for coating 3D prints
• Reynolds has hundreds of materials in stock
• 9 Locations throughout the USA
REYNOLDS ADVANCED MATERIALS
reynoldsam.com

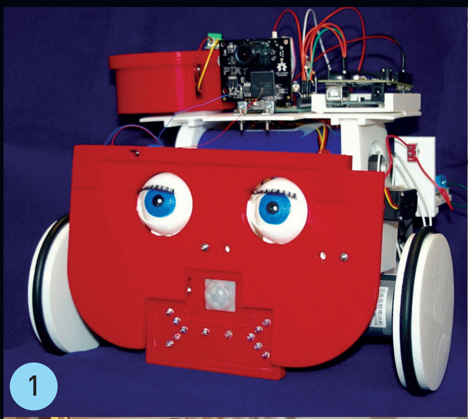

PLASTIC INJECTION MOLDING MACHINES - STARTING AT $595!
Mold your own plastic parts. Perfect for inventors, schools or companies. For details & videos visit our webpage.
The affordable Model 20A turns your workshop drill press into an efficient plastic injection molding machine. Simple to operate and it includes a digital temperature controller.
No expensive tooling is required - use aluminum or epoxy molds.
The bench Model 150A features a larger shot capacity and is perfect for protoyping or short production runs. Capable of producing up to 180 parts per hour!
We also carry:
* MOLDS
* CLAMPS
* ACCESSORIES
* PLASTIC PELLETS
MADE IN THE USA
PayPal
VISA
BANK
www.techkits.com
707-328-6244


THIS IS MU
ONLY SALE $25.00
VISION FOR EVERYTHING
SMART VISION SENSOR
ADVANCED VISUAL RECOGNITION SYSTEMS
- Based on embedded platform computer vision technique.
- Paired with Lego EV3, Power Functions, Hexbug, MU can recognize faces, follow people, track lines or balls. find the owner and say hello.
- Personalized greeting messages can be provided.
EFFICIENT LOCAL PERFORMANCE
- Fast boot and 2 hours battery life
HIGHLY EXPANDABLE
MU PCB Board II
- UART/SPI/I2C pins to interface with Arduino or Raspberry Pi
- Two sets of the 3-wire servo connectors
- IR / Bluetooth supported
- Voice input
- Low power consumption
http://store.morpx.com/
MU store accepts Visa, MasterCard, JCB and PayPal for online orders.
Sales / BD : sales@morpx.com
MORPX
VISION FOR EVERYTHING

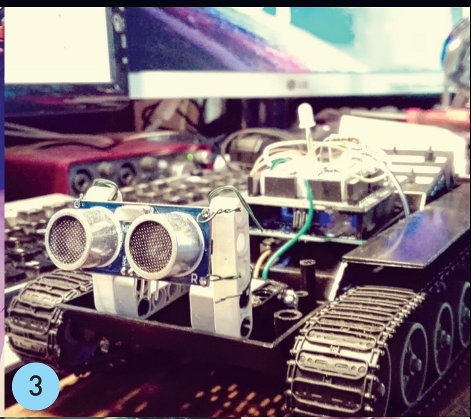
SHOW & TELL

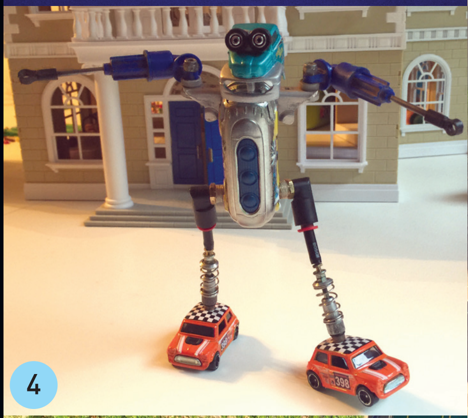
Dazzling projects from inventive makers like you

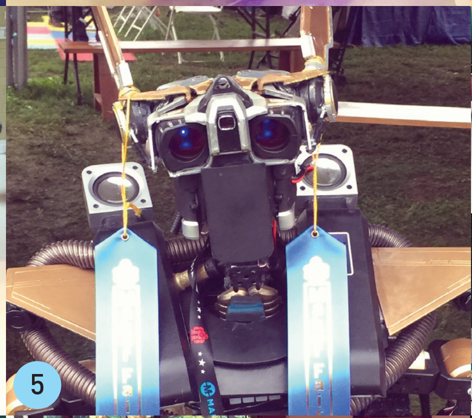
Sharing what you've made is half the joy of making! Check out these makers on Instagram, and show us your photos by tagging @makemagazine.


1


2


3

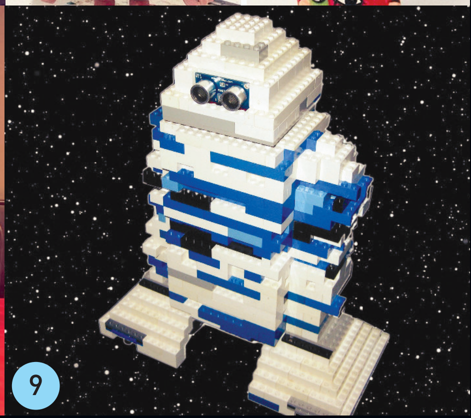

4


5


6

7

8


9

1 **Mike Rigsby**'s juicy little Mato bot is supercharged with 9,000 farads of capacitance, allowing her to run around talking, chasing a ball, and batting her lashes for 75 minutes on an 8-minute charge.

2 Reybotics CEO **Heriberto Reynoso** teamed up with the **Early Learning Foundation Academy** in South Texas to develop hardware "so that students in pre-K can colonize the surface of Mars (aka their playground)."

3 Engineering student **Ville Väisänen** (@villelectric) plans to upgrade TankBot (his first real robotics project) with a Raspberry Pi and OpenCV, but he's off to a great start so far using the Arduino Leonardo and his son's Lego Technics.

4 Junkbot strikes a pose to show off his spiffy roller skates for creator **Liam Robb**, who documents his die-cast car projects at @tippityplop.

5 **Ray Rumore**, age 9, built his traveling companion robot, Volt, with his dad over the last two years. Read more about Ray and Volt at makezine.com/go/volt-bot.

6 Not all robots run on wires and circuits! Artist **Gus Fink** (@gusfink) sculpts cute and creepy toys for sale at creeplings.com.

7 **Marc Genevat, Oriol Orra**, and **Juampe Pedro López** built this rugged Canyonero wagon, which runs on ROS, Raspberry Pi 3, Arduino, a motor shield, and rolls on second-hand longboard wheels.

8 This aluminum and edible duo (respectively, that is) is ready for a robot party with @instantpartypacks, thanks to the crafty finesse of maker **Kathryn Sartori**.

9 Using a Particle Photon, a few servos, and some toy bricks, **Sean McCormick** built and shared Brick Droid, a close cousin of R2-D2, on *Make:*'s Community Projects platform.


Maker Faire®
GREATEST
Show & TELL
ON Earth
MAY 19–21
12TH ANNUAL BAY AREA MAKER FAIRE
SAN MATEO EVENT CENTER
SEPT 23+24
8TH ANNUAL WORLD MAKER FAIRE
NEW YORK HALL OF SCIENCE
M
Maker Faire®
Global
FIND A FAIRE NEAR YOU
makerfaire.com

$9.99 US $11.99 CAN
ISBN: 978-1-68045-303-4
50999>
9 781680 453034